AL/CF-TR-1994-0077

ARMSTRONG LABORATORY

INTERIM TECHNICAL REPORT NO. 1 (ITRI)

Roger Andrews
Robert Baltzer
Andrew Boone
Lucy Garcia
Rick Gier
Bret Givens
Kathy Jackson
Medhat Korna
Ed Lehman
Cindy Martin
Evan Rolek
Michael Rountree
Ken Runner
Rod Sharp
Jim Stadler
Brett Storey
Mike Sweaney

VEDA INCORPORATED
5200 SPRINGFIELD PIKE, SUITE 200
DAYTON OH 45431-1265

DECEMBER 1993

INTERIM REPORT FOR THE PERIOD SEPTEMBER 1992 TO JUNE 1993

Approved for public release; distribution unlimited

AIR FORCE MATERIEL COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-7022

NOTICES

DISCLAIMER

TECHNICAL REVIEW AND APPROVAL

[illegible]

This report has been reviewed by the Office of Public Affairs (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

KENNETH R. BOFF, Chief
Human Engineering Division
Armstrong Laboratory

REPORT DOCUMENTATION PAGE

Form Approved
OMB No 0704-0188

1. AGENCY USE ONLY (Leave blank)

2. REPORT DATE
December 1993

3. REPORT TYPE AND DATES COVERED
Interim Report: Sep 1992 - Jun 1993

4. TITLE AND SUBTITLE
Interim Technical Report No. 1 (ITR1)

5. FUNDING NUMBERS
C: F33615-92-C-5936
PE: 63231F
PR: 2829
TA: 01
WU: 09

6. AUTHOR(S)
Andrews, R., Baltzer, R., Boone, A., Garcia, L., Gier, R., Givens, B., Jackson, K., Korna, M., Lehman, E., Martin, C., Rolek, E., Rountree, M., Runner, K., Sharp, R., Stadler, J., Storey, B., Sweany, M.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Veda Incorporated
5200 Springfield Pike
Suite 200
Dayton, OH 45431-1265

8. PERFORMING ORGANIZATION REPORT NUMBER
63819-93U/P60099

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Crew Systems Directorate
Armstrong Laboratory
Human Systems Center
Air Force Materiel Command
Wright-Patterson AFB, OH 45433-7022

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
AL/CF-TR-1994-0077

11. SUPPLEMENTARY NOTES
Subcontractor assistance: Brett Storey of Storey Consulting, Rocklin, CA.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

This report was prepared in two volumes. This is Volume I of Interim Technical Report No. 1 for the Crew Centered Cockpit Design (CCCD) Field Demonstration Program, Contract F33615-92-C-5936, Contract Data Requirements List (CDRL) Sequence Number A010. Volume I discusses technical accomplishments and results. Volume II contains supplementary material in the form of twelve appendices. (Volume II can be obtained by referring to Reference 66, Section 7.)

The objectives of the CCCD Field Demonstration Program are to upgrade, validate, and transition a new system for cockpit design. The system consists of a Crew-Centered System Design Process (CSDP) and a set of computer-aided tools known as the Cockpit Design System (CDS). This report summarizes accomplishments during the period September 1992 through June 1993 that include: development of an improved CSDP, system support and management; design of new tools to manage and trace design progress; conversion and improvements to existing software tools; and restructuring of real-time simulation software. The progress on the first of five Field Demonstrations, known as the F-16 Reconnaissance Mission, is described and includes information on planning, crew system analysis tasks, prototyping of design alternatives, and reconfiguration of the real-time simulator.

14. SUBJECT TERMS
CCADS, CATBALS, CCCD, CDS, COMBIMAN, CSDP, CPACR, DTM, FDSIM, F-16R, Field Demonstration No. 1, GIT, GMS, MDTOOL, QFD, SWAS, TMT, UNIX, VAX

15. NUMBER OF PAGES
146

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UNLIMITED

THIS PAGE LEFT BLANK INTENTIONALLY.

PREFACE

Interim Technical Report No. 1 (ITR1) for the Cockpit Design System (CDS), Crew-Centered Cockpit Design (CCCD) Field Demonstration Program, is submitted under United States Air Force (USAF) Contract F33615-92-C-5936, Contract Data Requirements List (CDRL) Sequence Number A010.

This document is the first interim technical report for the Crew-Centered Cockpit Design Field Demonstration Program, covering the period September 1992 through June 1993. It reports work performed for the Crew Systems Directorate, Armstrong Laboratory, Air Force Materiel Command, Wright-Patterson AFB, OH. LtCol Robert J. Collins and Major Julie Cohen served as the Project Managers, and Mr. Philip V. Kulwicki served as the Project Engineer. The work was performed by Veda Incorporated, Dayton, OH. Mr. Michael E. Rountree was the Veda Program Manager. This document is assigned Veda Document Number 63819-93U/P60099 and was prepared in two volumes. Volume I discusses technical accomplishments and results. Volume II contains supplementary material in the form of twelve appendices. Volume II was not published because it was not considered necessary for a complete understanding of the information contained in Volume I. If needed for in-depth study, Volume II can be obtained through the information given in Reference 66 of the Reference List (Section 7).

This report was accomplished with guidance from Mr. Michael Rountree and with contributions from Mr. Brett Storey of Storey Consulting, Rocklin, CA. It was compiled and edited by Ms. Katherine Jackson and Mr. Rodney Sharp, with technical assistance from Mr. Roger Andrews, Mr. Robert Baltzer, Mr. Andrew Boone, Ms. Lucy Garcia, Mr. Richard Gier, Mr. Bret Givens, Mr. Medhat Korna, Mr. Edward Lehman, Ms. Cynthia Martin, Mr. Evan Rolek, Mr. Kenneth Runner, Mr. James Stadler, and Mr. Michael Sweany.

THIS PAGE LEFT BLANK INTENTIONALLY.

VOLUME I
TABLE OF CONTENTS

VOLUME I
TABLE OF CONTENTS (Continued)

VOLUME I
TABLE OF CONTENTS (Continued)

VOLUME I
TABLE OF CONTENTS (Continued)

VOLUME I
LIST OF FIGURES

VOLUME I
LIST OF TABLES

VOLUME II
TABLE OF CONTENTS

ACRONYMS, TERMS, AND ABBREVIATIONS

A-A	Air-to-Air
A-CAL	Altitude Calibration
A-G	Air-to-Ground
A/S	Airspeed
AASPEM	Advanced Air-to-Air System Performance Evaluation Model
AB	Afterburner
ABCCC	Airborne Command and Control Center
ACCEL	Accelerate
ACM	Association of Computing Machinery
ACPCDB	Aircraft System and Crew Performance Data Base
ACQ	Acquisition
ADI	Attitude Direction Indicator
AED	Automated Experimental Design
AFB	Air Force Base
AFIT	Air Force Institute of Technology
AFMSS	Air Force Mission Support System
AFR	Air Force Regulation
AFSC	Air Force Systems Command
AGL	Above Ground Level
AHP	Analytical Hierarchy Process
AL/CFH	Armstrong Laboratory, Human Engineering Division
ALOW	Altitude Low
ALT	Altimeter
Anl	Analysis
ANSI	American National Standards Institute
Ant	Anticipated
AP	Apply Pressure
App	Application
ASC	Air Force Aeronautical Systems Center
ASCII	American Standard Code for Information Interchange
ASP	Analysis Study Plan
ASPJ	Airborne Self Protection Jammer
Assoc	Associated
ATARS	Advanced Tactical Air Reconnaissance System
ATHS	Automatic Target Hand-off System
AUTO	Automatic
AUX	Auxiliary
AWACS	Air Warning and Control System
BATS	Battle Area Tactical Simulation
BATS-SP	Battle Area Tactical Simulation Software Package
BIC	BSIM Interface Controller
BICMIO	BSIM Miscellaneous Input/Output
BIT	Built In Test
BRNG/DIST	Bearing/Distance
BRT	Brightness
BSIM	Breadboard Cockpit Simulator
C	Continuous
C-CADS	Crew-Centered Analysis and Design Support
C-SAINT	CAT-Systems Analysis of Integrated Network of Tasks

ACRONYMS, TERMS, AND ABBREVIATIONS (Continued)

C-TAC	CAT-customized TAC Brawler
C-TLAPREP	CAT-customized Timeline Analysis Preparation
C/D	Control/Display
C/O	Cut Out
CAD/CAM	Computer Aided Design/Computer Aided Manufacturing
CADSS	Cockpit Automation Design Support System
CALS	Computer-Aided Acquisition and Logistics Support
CASE	Computer-Aided Software Engineering
CAT	Cockpit Automation Technology
CATBATS	Cockpit Automation Technology Battle Area Tactical Simulation
CCB	Configuration Control Board
CCC	Cockpit Configuration Control
CCCD	Crew-Centered Cockpit Design
CCE	Cockpit Configuration Edit
CD-ROM	Compact Disk Read Only Memory
CDDT	Control and Display Development Tool
CDR	Critical Design Review
CDRL	Contract Data Requirements List
CDS	Cockpit Design System
CDT	Cockpit Design Team
CH/FL	Chaff/Flare
CI	Confidence Interval
CI	Configuration Item
CIPLP	Cockpit Instrument Panel Layout Program
CISC	Complex Instruction Set Computer
CM	Configuration Management
CMDB	Configuration Management Data Base
CML	Commercial
CMS	Code Management System
CMT	Concept Mapping Tool
COM	Communications
COM1	UHF Radio
COM2	VHF Radio
COMBIMAN	Computerized Biomechanical Man-model
COMP	Complete
CONT	Contrast
COORDS	Coordinates
CP	Change Proposal
CPT	Cockpit Product Tool
CPU	Central Processing Unit
CR	Change Request
CRAWL	Computerized Rapid Assessment Workload
CRS	Course
CRUS	Cruise
CSAT	Crew System Analysis Tools
CSCI	Computer Software Configuration Item
CSDE	Crew-Centered System Design Encyclopedia
CSDP	Crew-Centered System Design Process
CSDSDB	Crew System Design and Specification Data Base
CSEF	Crew Station Evaluation Facility

CSERIAC	Crew System Ergonomics Information Analysis Center
CSGT	Crew Station Geometry Tool
CSPDB	Crew System Performance Data Base
CSS	Crew System Specification
CSTGRF	C-SAINT Graphical (post processor program)
CSTPO	C-SAINT Preparation Zero Program
CSTSUM	C-SAINT Summary Post Processor Program
CTET	CAT customized Task Execution Time
CTETPREP	CAT customized Task Execution Time Preparation
CTL	Control
CTLA	CAT customized Timeline Analysis
CTLAP	CAT customized Timeline Analysis Plot
CTLAPREP	CAT customized Timeline Analysis Preparation
CTLAT	CAT customized Timeline Analysis Translator
Ctrl	Control
DAL	Data Accession List
DB	Data Base
DBMS	Data Base Management System
DCADS	Designer's Computer-aided Design System
DCO	Document Change Order
DCR	Documentation Change Request
DEC	Digital Equipment Corporation
DECCMS	DEC Code Management System
DECEL	Decelerate
DED	Data Entry Display
deg	degree
Dem/Val	Demonstration/Validation
DEN	Designer's Electronic Notebook
DEPR	Depression
DFAD	Digital Feature Analysis Data
DID	Data Item Description
DLA	Display Legibility Analysis
DLINK	Data Link
DMA	Direct Memory Access
DMCS	Data Management Control System
DOC	Document
DoD	Department of Defense
DoD-HDBK	Department of Defense Handbook
DoD-STD	Department of Defense Standard
DOF	Degree of Freedom
DRB	Design Review Board
DRD	Design Requirements Document
DT&E	Development Test and Evaluation
DTE	Digital Terrain Elevation
DTED	Digital Terrain Elevation Data
DTIC	Defense Technical Interchange Center
DTM	Design Traceability Manager
DTT	Design Trade-off Tool
DWB	Designer's Work Bench

E	East
E-Mail	Electronic Mail
E-VISION	External Vision model
ECO	Engineering Change Order
ECR	Engineering Change Request
EDN	Electronic Data News
EDSIM	Engineering Design Simulator
EED	Edit Event Definitions
EET	Edit Event Timeline
EFR	Event-Function Relationship
ENTR	Enter
EO	Electro-Optical
EQUIP	Equipment
ESQL	Embedded Structured Query Language
ETL	Event Timeline
EXA	Examine
EXCM	Execute Command
EXEC	Executive
ext	Extension
FAATAT	Function Allocation Trade Analysis Tool
FAR	Functional Analysis Report
FARIN	Functional Analysis Report Input
FATOOL	Function Analysis Tool
FCA	Functional Configuration Audit
FCR	Fire Control Radar
FDR	Flight Data Recorder
FEBA	Forward Edge of Battle Area
FFAT	Functional Flow Analysis Tool
FIXUISPDES	Fix for Print Destination Component of PRSCRN
FLIR	Forward Looking Infrared
FLOT	Forward Line of Troops
FLT	Flight
FMS	Full Mission Simulation
FORTRAN	Formula Translator
FPR	Function Procedure Relationship
fps	feet per second
ft	feet
FTL	Function Timeline
FWD	Forward
FY	Fiscal Year
GEOMB	Geometry Binary file
GIT	Geometry Interface Tool
GL	Graphics Libraries
GMS	Graphics Modeling System
GPS	Global Positioning System
GPX	Graphics Processing Extension
GSA	Government Services Agency
GUI	Graphical User Interface

ACRONYMS, TERMS, AND ABBREVIATIONS (Continued)

H/W	Hardware
HDG	Heading
Hi	High
HMD	Helmet-mounted Display
HOS	Human Operator Simulator
HOTAS	Hands-on Throttle And Stick
HP	Hewlett Packard
HSC	Human Systems Center
HSD	Horizontal Situation Display
HSI	Horizontal Situation Indicator
HUD	Heads-Up Display
HWCI	Hardware Configuration Item
Hz	Hertz
I-DEAS	Integrated Design Engineering Analysis Software
I/O	Input/Output
IADBX	Information Analysis Data Base
IAS	Indicated Air Speed
IATOOL	Information Analysis Tool
IBM	International Business Machines
ICD	Interface Control Document
ICP	Interface Control Panel
ICRAT	Information and Control Requirement Analysis Tool
IDE	Interactive Development Environment, Inc.
IDEF	Integrated Computer-aided Manufacturing Definition
IEEE	Institute of Electrical and Electronic Engineers
IFACE	Interface
IFF	Identify Friend or Foe
IGES	Initial Graphics Exchange Specification
IIOC	Intelligent Input/Output Control
INC/DEC	Increment/Decrement
Info	Information
INS	Inertial Navigation System
INSTRMNTS	Instruments
INT/EXT	Internal/External
Interp	Interpretation
INTRCPT	Interceptor
IP	Initial Point
IPD	Integrated Product Development
IPDT	Integrated Product Development Team
IR	Infrared
IRAD	Independent Research and Development
ITR	Interim Technical Report
IWSM	Integrated Weapon System Management
JSIPS	Joint Service Imagery Processing System
KNTS	Knots
LAN	Local Area Network
LAT	Latitude

ACRONYMS, TERMS, AND ABBREVIATIONS (Continued)

LH	Left Hand
LISP	List Processing
LLDB	Lessons Learned Data Base
Lo	Low
LOC	Line of Communication
LOC	Locate
LONG	Longitude
LSE	Language Sensitive Editor
LVL	Level
M-SEL	Mode Select
MAEO	Medium Altitude Electro-Optical
MATL	Material
Max	Maximum
MB	Megabyte
MCAD	Mechanical Computer-Aided Design
MCOS	Monte Carlo Operator Sample Generator
MD	Mode
Md	Medium
MDTOOL	Mission Decomposition Tool
MFD	Multifunction Display
MHz	Megahertz
MIL-H	Military Handbook
MIL-STD	Military Standard
Min	Minimum
MISC	Miscellaneous
MLTT	Mechanization Logic Tree Tool
MM	Millimeter
MMS	Mission Management System
Mngr	Manager
MON	Monitor
MPE	Mission Procedure Evaluation
MSA	Mission Scenario Analysis
MSADB	Mission Scenario Analysis Data Base
MSAIN	Mission Scenario Analysis Input (file)
MSAINT	Micro SAINT
MSAP	Major Systems Acquisition Process Phase
MSS	Mission Support System
MTA	Mission Timeline Analysis
MTM	Methods Time Measurements
MTP	Mission Task-Time Probability
N	North
NAECON	National Aerospace and Electronics Conference
NASA-TLX	National Aeronautical and Space Administration's Task Load Index
NAV	Navigation
NFS	Network File System
NM	Nautical Miles
NMT	Network Management Tool
NORM	Normal
NTSC	National Television Standards Committee

ACRONYMS, TERMS, AND ABBREVIATIONS (Continued)

O & M	Operation and Maintenance
OAR	Operator Assessment of Reach
OFP	Operational Flight Program
OPM	On-site Project Manager
OSB	Option Select Button
OSF	Open System Foundation
OT&E	Operational Test and Evaluation
OWLES	Operator Workload Evaluation System
PAWES	Performance Workload Evaluation System
PC	Personal Computer
PCA	Physical Configuration Audit
PCD	Performance/Control Display
PDR	Preliminary Design Review
PET	Procedure Execution Time
PETIN	Procedure Execution Time Input (file)
PFD	Pictorial Format Display
PIDS	Prime Item Development Specification
PK	Probability of Kill
PLRT	Polarity
PLTGGP	Plot via Graphical Package
PO	Purchase Order
POS	Position
POSIX	Portable Operating System Interconnect Extension
PP	Present Position
PPST	Program Planning Scheduling Tool
PPWEDB	Pilot Performance Workload and Error Data Base
Predic	Predicted
PRG	Program
PRGM	Program
PRIM	Primary
Proc	Procedures
PRSCRN	Print Screen
PRTT	Protocol's Requirement Traceability Tool
PTL	Procedure Timeline
PTR	Product Traceability Report
PTS	Part Task Simulation
PUP	Pop-up Point
PVI	Pilot Vehicle Interface
PWR	Power
QFD	Quality Function Deployment
QTY	Quantity
R&D	Research and Development
RAR	Radar Acquisition Range
RCCE	Reconnaissance
RCD	Record
RCL	Recall
RCS	Revision Control System
Recce	Reconnaissance

ACRONYMS, TERMS, AND ABBREVIATIONS (Continued)

REL	Release
RELGEO	Relative Geometry
Req	Requirements
RET	Reticule
RHAW	Radar Homing And Warning
RISC	Reduced Instruction Set Computer
RPM	Revolutions per Minute
Rslts	Results
RTN	Return
RTrace	Requirements Traceability Tool
RVRA	Runway Visual Range Analysis
RWE	Read/Write/Execute
RWR	Radar Warning Receiver
S-A	Surface-to-Air
S/N	Serial Number
S/W	Software
SAFE	Selected Areas for Evasion
SAINT	System Analysis of Integrated Network of Tasks
SAM	Surface-to-Air Missile
SCSI	Small Computer System Interface
SDRC	Structural Dynamics Research Corporation
SE	Standard Engine
SEQ	Sequence
SGI	Silicon Graphics Incorporated
Simapp	Simulation Application
SIMCLP	Simulation Control Logic Program
SIMWAM	Simulated Interactive Microcomputer Workload Analysis and Modeling
SL	Sherrill-Lubinski
SME	Subject Matter Expert
SMS	Stores Management System
SNSR	Sensor
SOI	Sensor-of-Interest
SON	Statement of Need
SORD	System Operational Requirements Document
SOW	Statement of Work
SPD	Special Purpose Display
SPEC	Specification
SPO	System Program Office
SQL	Structured Query Language
STC	Software Technology Conference
STP	Software Through Picture
STPT	Steerpoint
STSC	Software Technology Support Center
SUMAIN	Summary Main program
SUMMET	Survivable Measures Methods Evaluation Techniques
SW	Switch
SWAS	Sequitur's Workload Analysis System
SWAT	Subjective Workload Assessment Technique
SYM	Symbol

ACRONYMS, TERMS, AND ABBREVIATIONS (Continued)

T-ILS	TACAN - Instrument Landing System
TA	Terrain Avoidance
TACAN	Tactical Air Navigation
TAKE2	Tool for Automated Knowledge Engineering-2
TAMPS	Tactical Aircraft Mission Planning System
TAR	Threat Assessment Report
TBD	To Be Determined
TBMAIN	TAC Brawler Main Program
TCP/IP	Transmission Control Protocol/Internet Protocol
TEAMIN	Terrain Encounter Analysis Main program
TEAPLOT	Terrain Encounter Analysis Plot
TECH	Technical
TECHDB	Technology Assessment Data Base
TF	Terrain Following
TGT	Target
THROTL	Throttle
TIM	Technical Interchange Meeting
TLX	Task Load Index
TMT	Timeline Management Tool
TMU	Time-Measurement Unit
TPI	Technical Products Incorporated
TPS	Technical Publishing Software
TSD	Tactical Situation Display
TTL	Task Timeline
UARV	Unmanned Aerial Reconnaissance Vehicle
UFC	Up-Front Control
UHF	Ultra High Frequency
UNV	Universal
US	United States
USAF	United States Air Force
Val	Validation
VAPS	Virtual Avionics Prototyping System
VAX	Virtual Address Extension
VCR	Video Cassette Recorder
VDD	Version Description Document
VHF	Very High Frequency
VME	Virtual Memory Extension
VMIC	VME Microsystems International Corporation
VMS	Virtual Memory System
VOX	Vocal
VWS	VAX Windowing System
W/INDEX	Workload Index
WAM	Workload Assessment Monitor
WARN	Warning
WAT	Workload Analysis Tool
Wkld	Workload
WPAFB	Wright-Patterson Air Force Base
WPT	Waypoint

WSO	Weapons System Officer
WX	Weather
XED	Cross Edit
XMSN	Transmission
2D	Two-Dimensional
3D	Three-Dimensional
4GL	Fourth-Generation Language

1. INTRODUCTION AND SUMMARY

1.1 Background

The Crew-Centered Cockpit Design (CCCD) Field Demonstration Program (Reference 1) is the continuation of an advanced development project following eight years of dedicated work to conceive and improve a process and tools for cockpit design. This new process uses a systems engineering approach as a framework within which designers can focus more explicitly on crew capabilities and mission requirements. The Crew-Centered System Design Process (CSDP) is a structured, documented, and traceable design process. In its application, design decision rationale can be traced and used to avert and correct cockpit design flaws early in the development stage. The Cockpit Design System (CDS) is a set of procedures and computer-aided tools developed to assist in the design, analysis, and testing of various cockpit designs. The CSDP and the CDS are housed in the Armstrong Laboratory facility known as the Crew-Centered Analysis and Design Support (C-CADS) Laboratory, Building 248, Wright-Patterson Air Force Base, Ohio.

1.2 Objectives

The objectives of the Field Demonstration Program are to validate, upgrade, and support the transition of the CSDP and the CDS, including the application procedures and computer software, to both government and industry users. Validation will be attempted by invoking the CSDP and the CDS in selected cockpit design applications, primarily cockpit upgrades, for a variety of dissimilar operational aircraft systems. In addition, this effort includes assessing the needs of crew system designers; performing specific crew system analysis, design, and flight simulation tasks; implementing technical improvements for the existing CSDP and CDS; and promoting the use of the new technology in the Department of Defense (DoD) and DoD-contractor community.

1.3 Scope of Report

Interim Technical Report No. 1 (ITR1) describes the nature and results of the technical effort performed by Veda Incorporated (Veda) to support the Field Demonstration Program. It describes contract activity from 28 September 1992 to 11 June 1993, and satisfies the requirements of the Contract Data Requirements List (CDRL) A010. This report summarizes technical accomplishments and results, but does not attempt to describe detail to the lowest level. For example, input parameters and file transfer specifics for the first field demonstration, an F-16 Manned Reconnaissance Cockpit (F-16R), are summarized. The complete documentation of analysis, design, and test activities will be presented during periodic progress reviews and technical interchange meetings (TIMs) and will be reported in the Final Report (CDRL A013). At the completion of each field demonstration, separate reports that provide activity detail, validation test results, and upgrade recommendations will be available and will be placed on the Data Accession List (DAL).

1.4 Executive Summary

Prior to this contract, Veda provided support to the CCCD Project Office by assisting with the establishment of the C-CADS laboratory and by evaluating the CSDP and the CDS that were developed by the Boeing Company under an earlier Research and Development (R&D) contract. The technical approach to achieving the goals of the Field Demonstration Program was strongly influenced by two factors: (1) Veda was familiar with the CSDP and the CDS hardware and software, and (2) it was advantageous to accelerate the first field demonstration. Based on Veda's previous experience in rehosting and evaluating both the CSDP and the CDS, it was not necessary to formally assess needs for improvements, and much of the technical work was already in progress. At the outset of the contract, the two key areas of focus were: (1) improving the CSDP, and (2) restructuring the architecture of the Engineering Design Simulator (EDSIM). (The EDSIM is the part of the CDS that is intended to support cockpit evaluation through piloted simulation.)

General support, routine operations, and maintenance of the CSDP and the CDS was provided by having an on-site project manager (OPM) and a staff of software engineers and technicians. Daily support was responsive to both expected and unexpected requirements, such as short-notice demonstrations of the system. During numerous demonstrations, no significant system failures were experienced. Support included installation and maintenance of hardware and software, some of which require licenses and maintenance support agreements (Reference 66, Appendix A).

Configuration Management (CM) activity was continued from the previous contract, using the CM Plan that was previously developed and approved (Reference 6). In January 1993, a revision to the CM procedures was received from the government that somewhat simplified the CM process. Forms, data base, and procedures were modified accordingly by issuing an updated CM Plan (Reference 66, Appendix B).

As initial efforts in this Field Demonstration Program, research into new processes and tools for application to the CDS yielded several promising candidates. Quality Functional Deployment (QFD, Reference 2) and a related technique of the Analytical Hierarchy Process (AHP, Reference 3) were found to have value in evaluating alternative solutions, ranking and rating requirements, and performing tradeoff analyses. Concept Mapping (Reference 4) was found to be useful in eliciting expert information and organizing it for analytical decomposition. Sequitur's Workload Analysis System (SWAS, Reference 5) was used as a simplified and more generalized method for workload modeling. Each of the tools has an existing software package to assist in design problem applications, and each are being used in Field Demonstration No. 1.

Much of the technical work performed on the CDS was aimed at converting to a single operating system: UNIX. This conversion will enable the eventual elimination of the Virtual Address Extension/Virtual Memory System (VAX/VMS) operating system from the architecture, with the attendant cost savings. It will also migrate the CDS toward a more efficient, open architecture as compared to the more closed and proprietary VMS. The Graphical User Interface (GUI) known as UIM/X was selected and added to the system to facilitate the development of windowed applications through state-of-the-art information entry.

Significant progress was achieved in the design and development of software tools to satisfy the management and tracking requirements of the design process. The development of a new capability called the Design Traceability Manager (DTM) was recommended, requirements were identified, and development was started. The essential functionality of this new and powerful tool was demonstrated in May 1993.

Substantial progress was made in creating and documenting an improved CSDP. This streamlined version of the CSDP was directed at identifying the vital activities of effective cockpit design, and at defining procedures to accomplish those activities. It was developed by drawing upon the experience gained from recent cockpit design efforts (e.g., F-22) and by devising an orderly, logical flow of activities that: (1) met the periodic needs of the cockpit design team (CDT, an organized group of professionals who have experience in cockpit upgrades for specific aircraft, operations analysis, tactics and airmanship analysis, avionics, human engineering, and control and display engineering); (2) ensured that needed up-front analysis work is accomplished; (3) facilitated traceability of design and design decisions to mission requirements and crew capability; and (4) produced the needed reports and documentation for real-world programs. The first goal of the development effort was to produce a package for user/industry review. The new CSDP, often referred to as the CCCD Process, was documented (along with a scenario walk-through, a user questionnaire, and an evaluator questionnaire) for industry review (Reference 66, Appendices C, D, E, and F).

Selected CDS upgrades were implemented, as deemed appropriate, to reduce cost of ownership (e.g., eliminate the VAX/VMS) and to support Field Demonstration No. 1. Two Silicon Graphics Incorporated (SGI) workstations were added and memory and processor upgrades were made to other units. The Informix Data Base Management System (DBMS) was installed and is being used to develop many of the applications packages.

The CCCD application to convert selected F-16C aircraft into a tactical reconnaissance mission was selected as the subject of the first field demonstration. This is a real-world pilot-vehicle interface (PVI) problem that is likely to be the subject of development and testing during the next decade. The conversion of the EDSIM and the preparation for its evaluation is currently in progress. To make the EDSIM easy to reconfigure, it was necessary to restructure the simulation software. A layered architecture was developed that supports requirements for rapid prototyping of cockpit designs (Section 5.3.1.1). The software restructure was first accomplished using the existing Cockpit Automation Technology (CAT) Design cockpit, then converted to the F-16C for application to Field Demonstration No. 1. Analysis of mission requirements, system constraints, mission timelines, and task and workload analyses were performed as reported in Section 6 (Reference 66, Appendices G through L)

2. COCKPIT DESIGN SYSTEM SUPPORT

This section discusses CDS support activities that consist of management, planning, reporting, day-to-day operations, and Configuration Management.

2.1 Program Management

Numerous management meetings were held at the beginning of the contract to coordinate program direction and issues. After mutual agreement was reached on the content of Fiscal Year 1993 (FY93) work requirements, a two-part kick-off meeting was presented on 7-8 December 1992. The first session was held at Veda, and was open to individuals and agencies outside of the CCCD project. The second session was a working level meeting to lay out technical plans and activities relative to near-term goals.

The technical approach and contract schedule were formally changed by contract modification in February 1993. The changes represented the actions necessary to meet near-term program objectives. The principal change was to accelerate Field Demonstration No. 1 by approximately six months, and to adjust the other contract deliverables accordingly. The resultant overall program schedule is summarized in Figure 2.1-1.

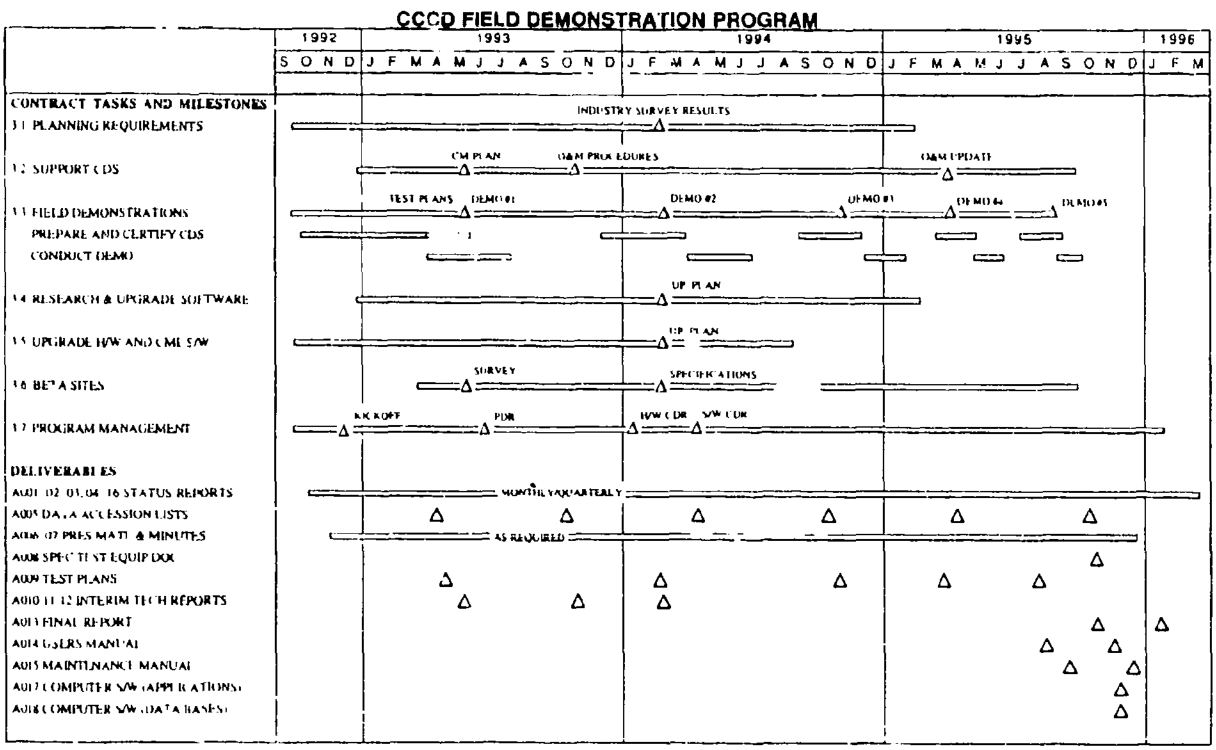


Figure 2.1-1. Overall Program Schedule

The F-16R was selected as the subject of the first field demonstration for the following reasons: (1) the conversion of an existing weapon system into a new tactical mission brings to the surface a number of PVI issues, such as downsizing of the crew from two (as in the RF-4 configuration), and the requirement for near-real-time, airborne intelligence data transmission; (2) the F-16R is a current problem representative of the type that is expected for the next decade and beyond; and (3) the F-16R PVI problem was also forecast for work in the Aeronautical Systems Center (ASC) Crew Station Evaluation Facility (CSEF) at Wright-Patterson Air Force Base (AFB). Therefore, performing this application under the CCCD Project has a potential for adding technical data and reducing risk for the subsequent CSEF evaluations.

2.1.1 Reporting

During the reporting period, the contract technical status, as well as financial, and performance information was provided through weekly progress reports, monthly status reports, quarterly status reports, and TIMs. In accordance with contract requirements, seven monthly status reports were submitted, two of which (January and April) incorporated additional quarterly reporting requirements.

2.1.2 Progress Review

A program review was held on 11 March 1993 at the Veda facility in Dayton, Ohio. Veda personnel, in conjunction with Storey Consulting, presented a general assessment of the program to-date; discussed the CSDP in terms of its advantages, development, and goals; reported the progress from Field Demonstration No. 1; and reviewed the software and hardware development for the CDS.

Discussion centered around the following topics: near-term success, parallel crew system activity, traceability, charting rules in the CSDP, introduction of the CSDP to industry for review, validation of the CSDP, node descriptions, Field Demonstration No. 1 schedule, conversion of the Cockpit Automation Technology Battle Area Tactical Simulation (CATBATS) to the F-16 flight model, sensor implementation and modeling, and Veda's approach to influencing the world of crew system design.

2.2 General Support and System Management

General support activities for the Field Demonstration program included tracking alternate and multiple versions of software, maintaining configuration management records, demonstrating CDS components and system functions, developing or modifying software, administering CDS data bases, incorporating CDS upgrades, and providing maintenance and licensing support. The revised Maintenance and Licensing Agreement reflects the most current commercial software configurations (Reference 66, Appendix A).

Daily support of the C-CADS laboratory was performed using responsive management techniques and flexibility to support unexpected requirements, such as short-notice demonstrations. These demonstrations were performed in parallel with, and without significantly interrupting, ongoing project and system management activities. In addition, daily support was provided for the installation of hardware and software, for the operation and maintenance of the C-CADS, and for the assessment and verification of the CDS components, including new and old versions of many components.

2.2.1 Crew-Centered Analysis and Design Support Laboratory

The C-CADS laboratory was originally configured as one large open area. This setup was not conducive to concurrent software development activities, and the conduct of simulator demonstrations, government visits, and industry walk-throughs. To enhance the professional environment and to better isolate activities, the laboratory was reconfigured into the following three areas: (1) the Design and Analysis Area, (2) the EDSIM Area, and (3) the Support Area. The laboratory is shown in Figure 2.2.1-1.

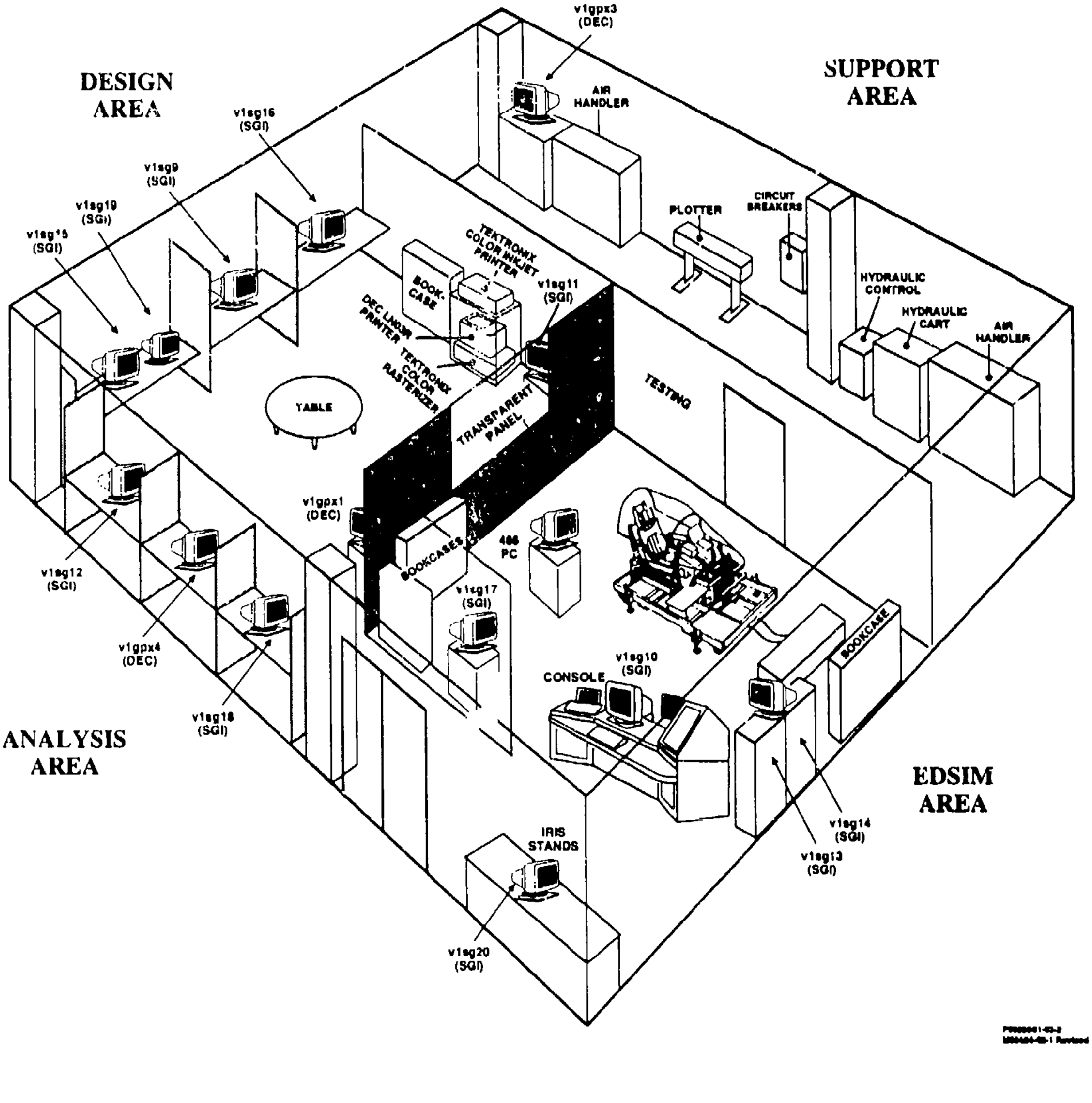


Figure 2.2.1-1. C-CADS Laboratory

The Design and Analysis Area consists of nine workstations, a color inkjet printer, a color rasterizer, and an LN03R postscript printer. Seven of the nine workstations (v1sg9, v1sg11, v1sg12, v1sg15, v1sg16, v1sg18, and v1sg19) are SGI products. The other two workstations (v1gpx1 and v1gpx4) are Digital Equipment Corporation (DEC) VAX Station II Graphics Processing Extension (GPX) products.

The EDSIM Area consists of a cockpit simulator, a console, a 486 Gateway personal computer (PC), and five SGI workstations (v1sg10, v1sg13, v1sg14, v1sg17, and v1sg20), a video camera, video tape recorder, and a video monitor.

The Support Area consists of a DEC workstation (vlgpx3), two air handlers, a plotter, a circuit breaker box, a hydraulic control, and a hydraulic cart.

In addition to the above areas, the C-CADS laboratory houses a DEC VAX 8700 host computer system and a line printer in an adjoining room.

2.2.2 Maintenance of Commercial Software and Hardware

Maintenance support services for the CDS commercial software and hardware components were acquired for the following commercial products: Informix, Integrated Design Engineering Analysis Software (I-DEAS), Graphics Modeling System (GMS), Mission Decomposition Tool (MDTOOL), CATBATS, SWAS, DI-3000, and QFD.

Management of the contractual vehicles for maintaining the commercial components of the CDS included the following:

a. Contacted the maintenance contractors to report hardware and software problems and to obtain scheduled maintenance.

b. Obtained the newest software releases for GMS, MDTOOL, and CATBATS.

c. Handled problems with SGI equipment and DEC equipment through maintenance agreements. During this reporting period, eight major hardware problems and eleven minor software problems were encountered. Two SGI monitors and several cables were replaced at no additional cost.

d. Worked with maintenance contractors through telephone support to further analyze and correct problems.

e. Obtained other commercial products, such as QFD and SWAS, that satisfy specific requirements. In the case of SWAS, a special agreement was reached that saved the initial purchase price and resulted in only a maintenance fee.

f. Restructured maintenance agreements for I-DEAS, GMS, MDTOOL, and CATBATS that cover long periods of time at no additional cost.

g. Obtained an additional SGI platform with four processors and total memory of 128 megabytes (MB). Two of the processors and 64 MB of memory were acquired at no cost (see Section 5.1).

h. Obtained DI-3000 for evaluation and conversion of the CDS components from the VAX environment to the UNIX environment at no cost.

2.2.3 Maintenance of Custom Cockpit Design System Components

Several CDS components, such as MDTOOL, CATBATS, GMS, and I-DEAS, are commercial products that are customized to interface with various other commercial products. Each vendor was consulted to ensure that current system functions and future enhancements to the CDS remain functional. By working closely with the vendors, Veda was able to maximize the use of commercial product enhancements at no additional cost to the project.

The Field Demonstration No. 1 displays were developed using GMS and were sent to Sherrill-Lubinski (SL) for optimization and improvement of the update rate (Section 5.3.7). I-DEAS was originally purchased for the VAX 8700. The version needed for the Silicon Graphics platform was acquired from Structural Dynamics Research Corporation (SDRC) for a minimum fee rather than purchasing a new version. Merit Technology, Incorporated extended the manhours on the maintenance agreement rather than limiting it to one year. These hours were used to develop enhancements to both MDTOOL and CATBATS (see Sections 5.3.3 and 5.3.6). The IRIX operating system 4.0.4 was installed on the Silicon Graphics machines. MDTOOL 4.06 and CATBATS 5.33 were compiled under IRIX 4.0.4 by Merit Technology, Incorporated.

2.2.4 Crew-Centered System Design Process

The CSDP provides the framework for the application of the CDS tools. The CDS tools within the CSDP framework will directly support specific crew system analysis, design, and evaluation activities in a systematic traceable flow. Considerable progress was made in creating and documenting an improved CSDP, and in achieving the goal of housing and managing it in the CDS, using the Informix DBMS. (Note: The original contract-delivered process is referred to as the Crew-Centered System Design Encyclopedia (CSDE); the new process is referred to interchangeably as the CSDP or the CCCD Process.)

During the transition from the CSDE to the CSDP, the management of both processes was established within the DTM (Section 5.3.2) environment so that the user could either reference the CSDE for further information or use the CSDP to actually guide cockpit design activities. The Methodology Data Base was developed in the Informix DBMS to manage the contents of both the CSDE and the CSDP. The structure of the Methodology Data Base allows the definition of four distinct tables: activities, procedures, technicals, and information pages. The activities table contains an overview description of its referenced activity (including a listing of procedural steps involved in performing the activity). The procedures table contains specific data for the accomplishment of each required procedure. The technicals and information pages tables contain additional information on the activity and the product that results from the performance of the activity.

The baseline CSDP was developed, implemented, and managed using Microsoft Project software. The data was then electronically transferred to the Informix DBMS. Several X-Windows application programs were developed to maintain and manage the CSDP (Sections 5.3.2.5 to 5.3.2.8).

2.2.5 Cockpit Design System Management

The successful management of the CDS resulted in the implementation of the CSDP and the smooth and near failure-free operation of the entire C-CADS laboratory, which was operational at all times. The development of new components and the enhancement of the existing ones continued throughout the reporting period.

2.2.6 Demonstrations of the Cockpit Design System

Over the course of this contract, several demonstrations of the CDS were supported. These demonstrations exercised the CDS and components. The demonstrations lasted between one and two hours and included an introductory briefing to present the objectives, and an overview of the demonstration to observers who were not familiar with the CDS. Some demonstrations were primarily walk-throughs where only portions of the capabilities were shown, others displayed the full capabilities of the current system. The demonstrations were performed without interrupting either an on-going application or the system maintenance activities.

2.3 Configuration Management

This section describes the current status and progress of the CM of the CDS. The objectives of CM are to track the status of the current configuration, and to provide a systematic means of tracking problems and suggested improvements from initial discovery through final disposition. Throughout this process, CM provides the CCCD Program Office with visibility into the evolving configuration of the CDS developmental and product baselines. The baselines include software, hardware, process, and documentation. The CM system also provides a means of tracking the inventory, maintenance, and license agreements.

Under Delivery Order 9 of the previous CCCD support contract, the *Configuration Management Plan for the C-CADS Laboratory* (Reference 6) was prepared and delivered. This original plan specified procedures for CM and also specified requirements for a CM DBMS. The CM DBMS was developed using the R:Base commercial data base management system hosted on an IBM-compatible PC. The CM DBMS was then populated with information pertaining to the CDS hardware, software, and media inventory; maintenance and license contracts; Engineering Change Requests (ECRs); Documentation Change Requests (DCRs); and Engineering Change Orders (ECOs).

2.3.1 Accomplishments

An objective assessment of CM-related activities yielded the following accomplishments:

a. Veda completed an extensive audit of the CDS inventory. This inventory consists of several thousand records describing the identification, location, and status of all CDS hardware, software, and media items. In addition to the audit, the CM DBMS was queried on several occasions to locate items in the inventory, and the necessary records were found to be present. Also, the accuracy of the descriptive information in many of the records was improved.

b. The maintenance and license information contained in the data base was helpful in preparing recommendations for FY94 maintenance and licensing (Reference 66, Appendix A), but was not sufficient. The stored data describes existing maintenance and license contracts, and therefore must be supplemented by vendor's quotes when preparing a plan for future coverage. The information must also be updated to include the newly-obtained configuration items.

c. Seven Configuration Control Board (CCB) meetings and approximately twenty-five Design Review Board (DRB) meetings were conducted.

d. Approximately 300 ECRs were completed by the CDS team and other CDS users, and entered into the CM DBMS. The DBMS reporting features provided an adequate means of tracking these ECRs, listing them by status, configuration item (CI) name, and assignee. However, there were some limitations in the CM DBMS arising in some cases from R:Base features, and in other cases from partial implementation of CM-supporting functions. Specifically, only one CI can be cited in a given ECR. This prevents the creation and tracking of change requests that may impact more than one software or hardware item. Moreover, the technique used to designate the CI is awkward: the title of the ECR must be prefaced with the CI name followed by a space, dash, and space, e.g., MDTOOL - Fails to accept sufficient number of event types. In addition, R:Base provides very limited text-editing and text-formatting capabilities. Material cannot be tabulated, underlined, or scanned for character strings.

2.3.2 Improvements

During the reporting period, several areas of improvement for the CM procedures and tools were noted. These areas involved: (1) the establishment of regular DRB and CCB meetings; (2) the processing, approval, and completion cycle of ECRs, ECOs, and DCRs; (3) the use of the Class I/Class II categorization scheme for ECRs; and (4) the content of the inventory data records that were downloaded from the original files.

On 7 January 1993, the CCCD Program Office released a memorandum entitled *CDS Configuration Control Program* that outlined changes to improve the existing CM procedures. Based on the memorandum, the following changes were made or are being made:

a. ECRs were replaced by Change Requests (CRs), which could address any requested change (process, hardware, software, or documentation). The CR is similar to the ECR, but covers requested changes to any or all of the following: CSDP, CDS documentation, CDS tools, or CDS hardware items. The CR includes information about the background of the submitter and the context in which the change was requested.

b. Change Proposals (CPs), rather than ECOs, are now prepared to delineate a recommended approach to the solution of one or more related CRs. The format for the CP varies substantially from the former ECO. It now includes a discussion of the relationship of the proposed change to the design process, a schedule of activities, the impact on other work, and a discussion of the likelihood of successful completion.

c. Three types of meetings will be held on a regular basis: TIMs, DRB meetings, and CCB meetings. TIMs, which constitute a new opportunity for joint Veda/CCCD Program Office coordination, will be held frequently as informal forums for the discussion of alternative approaches to CRs. TIMs will permit greater Air Force involvement in the hardware/software change process. DRB meetings will be held on a biweekly basis to discuss the status of new and ongoing CPs. CCB meetings will be held monthly to prioritize CPs and to approve selected CPs.

d. The Class I/Class II distinction for CPs will be applied on a more consistent basis. Any CP that requires more than 40 labor-hours to complete or entails a purchase amount of more than $2000 will be classified as a Class I CP. As such, it will be presented to the CCB for prioritization and approval before beginning work. Class II CPs will continue to be subject to the approval of the CCCD OPM.

e. All CRs and CPs will be discussed at DRB meetings and logged into the CM DBMS. While the CR can be entered in its entirety, inherent limitations in the R:Base system mitigate against full implementation of the CP form in the CM DBMS. First, the CPs tend to be lengthy, consisting of sixteen scrolling textual fields. When several completed CPs were entered into the system, the latter fields were truncated, apparently due to the inability of R:Base to handle tables of this length. A possible solution is to constrain the field length, but this would be at cross purposes to the objective to have detailed, self-explanatory proposals. Second, several of the CPs include figures, which greatly assist in explaining the recommended software or hardware architecture, functional flow, etc. However, R:Base cannot store graphical information, so these items cannot be retained with the completed CP. A third limitation is the inability of R:Base to handle tabulated fields of data. For example, if an engineer wanted to provide a table of performance parameters for several alternative solutions, he/she could not do so in R:Base. Fourth, entering the textual data into R:Base is difficult due to the absence of a full-featured text editor. For example, there are no search, replace, or text-formatting capabilities.

It should be noted that these limitations are characteristic of most commercial DBMSs and are not specific to R:Base. R:Base was and still is a good choice for managing multiple relational tables of alphanumeric data. Neither R:Base nor any other known commercial DBMS can handle the data base of integrated textual and graphical material that would be associated with full on-line CP storage.

In subsequent meetings it was agreed to retain only summary data for the CPs in on-line form using R:Base. The CP Summary Record will allow Veda's CM personnel to track the status of the CPs and their related CRs. The full content of the CPs, including text and graphics, will be stored on the Macintosh documentation workstation located in the Veda on-site office. A unique chronological number will be assigned to each CP on the documentation workstation and also entered into the R:Base CP Summary Record, to support tracking and retrieval. This will provide a viable and efficient solution to CP-tracking needs. Additionally, the availability of the CP information on the documentation workstation will allow Veda to incorporate the text and graphics into CDS documents, such as ITRs, Users Manuals, and Programmers Manuals.

Following implementation of the changes described above, ten CPs were prepared and several DRB meetings were held to discuss them. The current status is as follows:

CP1: Geometry Interface Tool (GIT). Approved and in progress.

CP2: DTM. Pending United States Air Force (USAF) review of the DTM Design Document, which is currently being completed.

CP3: Timeline Management Tool (TMT). Pending USAF review of the TMT Design Document, which is currently being completed.

CP4: EDSIM restructuring. Approved and in progress.

CP5: VMS-to-UNIX conversion. On hold pending further detail on plans.

CP6: Additional SGI workstations. Approved. New workstations have been installed.

CP7: QFD Designer. On hold pending evaluation of QFD/AHP.

CP8: Merit Support. Deleted; CP not necessary. CPs are submitted at conclusion of Merit modifications to software.

CP9: Concept Interpreter. On hold pending evaluation of the Tool for Automated Knowledge Engineering-2 (TAKE2).

CP10: SWAS. On hold pending USAF determination of whether a CP is needed.

CCB meetings were not held for the above actions, because the DRB interfaced with CCB personnel to determine each status.

2.3.3 Configuration Management Plan Update

The basis for the revised CM plan (Reference 66, Appendix B) is the procedural approach (Figure 2.3.3-1) that was outlined in the memorandum of 7 January 1993 from the CCCD Program Office.

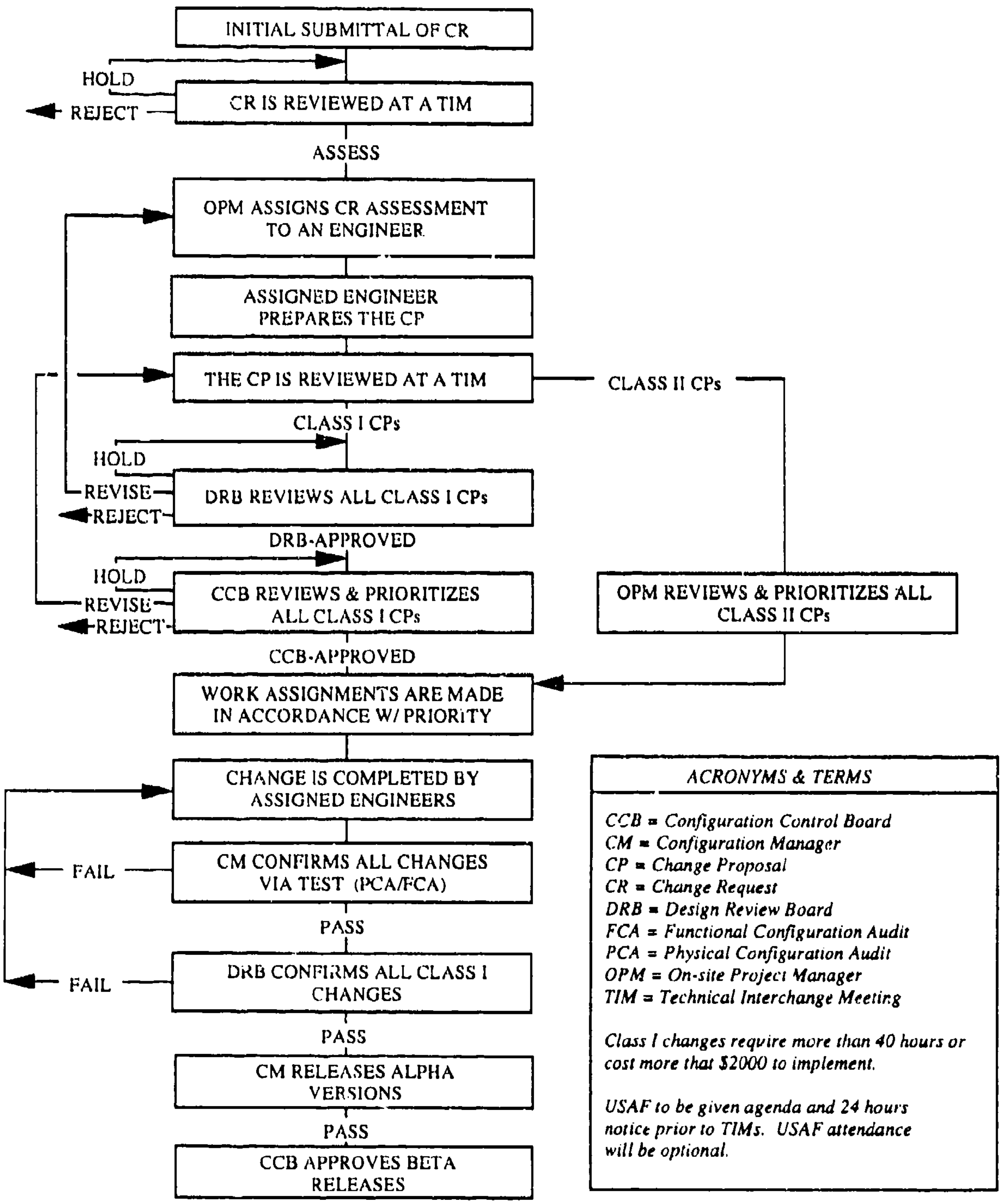


Figure 2.3.3-1. CM Procedural Approach

The revised CM procedure begins with submittal of the CR by a CDS user, a member of the CCCD Program Office, or a member of the CDS support team. The submittal process will be accomplished initially by the delivery of a handwritten or typewritten CR form. In the f 'ture, submission will be done electronically by providing personnel with limited access to the CM DBMS and by assigning a consecutive number to each CR at the time of submittal.

New CRs will be reviewed at regular TIMs. The CDS OPM will schedule and chair the TIMs. The TIMs will be attended by the CM Manager and by members of the CDS support team who are knowledgeable in the area affected by the CR. The CCCD Program Office will be given at least a 24-hour notice prior to each TIM, and Air Force participation wi be optional. During the review, each CR will be given a status: *Hold, Reject,* or *Assess.* The CRs classified as *Hold* will be reviewed at the next TIM. This action will be taken when a CR requires further detail, or time does not permit its evaluation at a given TIM. The CRs classified as *Reject* will be those that are not sufficiently clear or are due to operator error. The CRs classified as *Assess* will be those awaiting assignment to an engineer for the preparation of a CP.

The assigned engineer will prepare a CP to identify the required change to the CDS hardware, software, or process. When completed, the CP will be reviewed at another TIM. Each CP will be classified by the OPM as Class I or Class II; Class I CPs require more than 40 hours or cost more than $2000 to implement.

As shown in Figure 2.3.3-1, the procedures differ between Class I and Class II CPs. Class II CPs, by virtue of their limited magnitude, will be reviewed and prioritized by the OPM according to available resources and schedule.

In contrast, Class I CPs will be reviewed by the DRB to determine completeness, clarity, and consistency with long-range CDS goals and objectives. At this point the DRB can hold, reject, or approve CPs for further processing. On-hold CPs will be reevaluated at the next DRB meeting. Rejected CPs will be routed back to the OPM for reassignment and/or rework by engineering personnel.

DRB-approved Class I CPs will be reviewed and prioritized at a CCB meeting. The CCB can hold, reject, or approve CPs for further processing. On-hold CPs will be reevaluated at the next scheduled CCB meeting. Rejected CPs will be sent back to the DRB for reassessment.

Class II CPs and CCB-approved Class I CPs will be assigned to CDS personnel for implementation according to pric ity and schedule constraints. Necessary changes to the hardware, software, process, and/or documentation will be made by the assigned personnel.

The CM Manage or designee will confirm the completeness and correctness of changes (both Class I and Class II) in a Physical Configuration Audit (PCA) and Functional Configuration Audit (FCA). The FCA, which consists of a test of the functionality of the modified configuration item, will be conducted prior to release of the Alpha version of the item. The PCA, which consists of a verification that the item is in the proper physical configuration (i.e., that the source code has been stored in the proper locations, that the documentation has been updated to reflect the change and has been properly stored, etc.) will be conducted by the CM Manager or designee. All Class I changes will be reviewed by the DRB in meetings that will take place biweekly, depending on the existence of CRs and CPs to be reviewed. Incorrect or incomplete changes will be returned to the assigned personnel for correction.

Following successful completion of a modification to the CDS hardware, software or process, the modified items will be released as Alpha versions. These versions will be used only by CDS personnel until full confidence in their functionality is achieved.

When a plateau in capabilities has been achieved (for example, when the CSDP tools and documentation have been modified and verified to provide a set of integrated and comprehensive capabilities), the OPM will ask the CCB to approve the release of these items as Beta versions. New versions of configuration items will not be released on a piecemeal basis; rather, they will be released only in logical sets that, together, provide a new level of CDS capabilities. The CCB will review the proposed Beta releases. Only approved items will be released to Beta sites and other CDS installations.

2.3.4 Configuration Management Data Base

The CM DBMS was implemented initially to satisfy the requirements stated in the first version of the CM Plan. This included inventory control and tracking functions for ECRs, DCRs, ECOs and DCOs. The CM DBMS (Section 3.1.3) is now being modified to replace all forms and tables related to ECRs, ECOs, and DCRs with CRs and CPs. The data and procedures are being brought into agreement with the revised CM procedures.

The new CR form contains many new data fields that were not present in the ECR form that was implemented in the initial CM DBMS. Table 2.3.4-1 identifies the data fields that are contained in the CR. A definition of each field is provided, including a specification of the length and content. The next six columns contain the read/write privileges to be given to the CR submitter, the CDS users, the CDS hardware/software support personnel, CM personnel, CDS project management personnel (the Program Manager and the OPM), and the CCCD Program Office Personnel. Table 2.3.4-2 provides the same information for the CP.

Modifications to the existing CM DBMS to support the new CR and CP forms were initiated during the reporting period, but were not completed. The forms were created for data entry and CP report procedures were written, but the reports were unacceptable due to limitations in R:Base, the commercial DBMS in which the CM DBMS is implemented. Specifically, R:Base will not accommodate a full CP record, which contains many large data fields. After reaching its maximum record limit, R:Base truncates the large textual fields to only a few characters each. When this limitation was encountered, work on other functions was suspended. Linked lists of related CRs were not incorporated into the CP form, nor were links to the assigned personnel made. The recommended and recently approved solution to this problem is to store only CP summary data on-line, and to store the entire CP record on the Macintosh documentation Workstation. In the far term, the feasibility of rehosting the CM DBMS in Informix on an SGI workstation will be investigated as a means of providing on-line, multi-user access.

All Configuration Software Configuration Items (CSCIs) and Hardware Configuration Items (HWCIs) that have been created or acquired in preparation for Field Demonstration No. 1 will be given CSCI or HWCI numbers and identified as items in the CDS developmental configuration. The developmental configuration comprises the software, hardware, and associated technical documentation that define the evolving configuration of the CDS during development. Table 2.3.4-3 identifies the major components of the current CDS developmental configuration.

Entry of new configuration items into the CDS baseline is contingent on the completion of development, documentation, and testing. The original baseline CDS system used CSCI numbers 1 through 181. For new CSCIs developed since the delivery of the system to the USAF, the following CSCI-numbering scheme is recommended:

Table 2.3.4-1. Fields and Associated Read/Write Privileges for a Change Request Record

DATA FIELD	DEFINITION	OPTIONAL (O) OR REQUIRED (R)	CR SUBMITTER	OTHER CDS USERS	CDS HW/SW PERSONNEL	CM PERSONNEL	PROJECT MANAGE-MENT	CCCD PERSONNEL
Date	Date of initial submission. Initialized to date of CR record creation by the submitter.	R	Read (autogen)	Read	Read	Read/Write	Read	Read
Name	Name of submitter. Selectable from a predefined list of CDS users, or entered manually.	R	Read/Write	Read	Read	Read/Write	Read	Read
Organization	Initialized based upon submitter's name.	O	Read/Write	Read	Read	Read/Write	Read	Read
Telephone Number	Initialized based upon submitter's name.	O	Read/Write	Read	Read	Read/Write	Read	Read
CI Name	Name of hardware or software item for which change is being requested. Selected from a predefined list of CIs.	R	Read/Write	Read	Read	Read/Write	Read	Read
CI Version Number	Version number of the CI that the submitter was working with.	O	Read/Write	Read	Read	Read/Write	Read	Read
User Experience Level	One of three levels: novice, intermediate, or expert. Selected from a predefined list by the submitter.	O	Read/Write	Read	Read	Read/Write	Read	Read
Short Description	Descriptive title of CR. 50-character maximum length	R	Read/Write	Read	Read	Read/Write	Read	Read
Impact	Criticality rating of 1 to 5, where 1 = operational failure, 5 = minor annoyance. Selected from a predefined list by the submitter.	R	Read/Write	Read	Read	Read/Write	Read	Read
General Summary	A textual description of the problem and/or the suggested change. Maximum length: 2000 characters	R	Read/Write	Read	Read	Read/Write	Read	Read

Table 2.3.4-1. Fields and Associated Read/Write Privileges for a Change Request Record (Continued)

DATA FIELD	DEFINITION	OPTIONAL (O) OR REQUIRED (R)	CR SUBMITTER	OTHER CDS USERS	CDS HW/SW PERSONNEL	CM PERSONNEL	PROJECT MANAGEMENT	CCCD PERSONNEL
Work Context	A textual description of the manner in which the CI is used when the problem occurs or the suggested improvement is applicable. Maximum length: 2000 characters.	O	Read/Write	Read	Read	Read/Write	Read	Read
Other CIs Affected	Names of other hardware or software items which the requested change might impact. Selected from a predefined list of CDS CIs.	O	Read/Write	Read	Read	Read/Write	Read	Read
Suggested Solution	A textual description of the possible ways in which the problem or suggestion might be handled. Maximum length: 2000 characters.	O	Read/Write	Read	Read	Read/Write	Read	Read
CR Number	Integer number assigned to CR. Autogenerated by CM system when CR is created, but able to be overridden by CM personnel. Range: 1 to 9999.	R	Read (autogen)	Read	Read	Read/Write	Read	Read
CR Status	Statuses are selected from a predefined list of CR status values: Hold, Rejected, TIM-Approved, or Completed, with the effective date of each status.	O	Read	Read	Read	Read/Write	Read	Read
CP Number	Number of the CP (if any) that proposes a solution to this problem or requested change. (Multiple CRs may be addressed by one CP.)	O	Read	Read	Read	Read/Write	Read	Read

Table 2.3.4-2. Fields and Associated Read/Write Privileges for a Change Proposal Record

DATA FIELD	DEFINITION	SUBMITTER	OTHER CDS USERS	OTHER CDS HW/SW PERSONNEL	CM PERSONNEL	PROJECT MANAGE-MENT	CCCD PERSONNEL
CP Number	Integer number assigned to CP by CM system (autogenerated). Range: 1 to 9999.	Read	Read	Read	Read/Write	Read	Read
CP Title	Descriptive title of CP. 50-character maximum length.	Read/Write	Read	Read	Read/Write	Read	Read
CP Version Number	Numerical identifier of CP version. Range: 1 to 99.	Read/Write	Read	Read	Read/Write	Read	Read
Date	Initialized to date of initial preparation. Can be updated to indicate date of latest version.	Read/Write	Read	Read	Read/Write	Read	Read
Cost	Summary of the cost of the proposed work, broken down by labor hours and purchases.	Read/Write	Read	Read	Read/Write	Read	Read
Objectives	Textual description of the purpose of the proposed change. Maximum length: 4000 characters.	Read/Write	Read	Read	Read/Write	Read	Read
Relationship to the Design Process	Textual description of the ties between the cockpit design process and the proposed change. Maximum length: 4000 characters.	Read/Write	Read	Read	Read/Write	Read	Read
Products	Textual description of the ties between the cockpit design process and the proposed change. Maximum length: 4000 characters.	Read/Write	Read	Read	Read/Write	Read	Read
Related CRs	A tabulated list of numbers and titles of one or more (limit 99) Change Requests related to this Change Proposal. Selected from the table of CRs resident in the data base.	Read/Write	Read	Read	Read/Write	Read	Read

Table 2.3.4-2. Fields and Associated Read/Write Privileges for a Change Proposal Record (Continued)

DATA FIELD	DEFINITION	SUBMITTER	OTHER CDS USERS	CDS HW/SW PERSONNEL	CM PERSONNEL	PROJECT MANAGE-MENT	CCCD PERSONNEL
Assigned Personnel	Names of personnel who will implement the change. Selected from a predefined list of CDS personnel.	Read	Read	Read	Read/Write	Read	Read
Schedule	Milestones and any other schedule-related data. Textual. Maximum length: 500 characters.	Read/Write	Read	Read	Read/Write	Read	Read
Cost	Estimated costs (item, cost, maintenance cost, vendors). Textual. Maximum length: 500 characters.	Read/Write	Read	Read	Read/Write	Read	Read
Other CIs Affected	Names of other hardware or software items which the requested change might impact. Selected from a predefined list of CDS CIs. Textual. Maximum length: 500 characters.	Read/Write	Read	Read	Read/Write	Read	Read
Impact on Other Work	Any conflicts with other planned or ongoing work. Textual. Maximum length: 2000 characters.	Read/Write	Read	Read	Read/Write	Read	Read
Likelihood of Successful Completion	Discussion of risks and risk management. Maximum length: 2000 characters.	Read/Write	Read	Read	Read/Write	Read	Read
Alternative Approaches	Description of other possible solutions. Refer to attached tradeoff studies as needed. Textual. Maximum length: 2000 characters.	Read/Write	Read	Read	Read/Write	Read	Read
Test Requirements	Verification approach. Textual. Maximum length: 500 characters.	Read/Write	Read	Read	Read/Write	Read	Read

Table 2.3.4-2. Fields and Associated Read/Write Privileges for a Change Proposal Record (Continued)

Documentation Requirements	Necessary changes or additions to CDS documentation. Textual. Maximum length: 500 characters.	Read/Write	Read	Read	Read/Write	Read	Read
Class	I or II.	Read	Read	Read	Read/Write	Read	Read
Priority	1 (highest) through 5 (lowest).	Read	Read	Read	Read/Write	Read	Read
Status	Statuses are selected from a predefined list: Hold, Rejected, DRB Approved for Work, CCB Approved for Work, In Work, PCA/FCA, DRB Confirmation, Alpha Release, Beta Release. Effective dates should be paired with each status.	Read	Read	Read	Read/Write	Read	Read

Table 2.3.4-3. Developmental Configuration of the CDS Tools

FUNCTIONAL REQUIREMENT	CDS TOOL
Manage process, tools, project implementation, and traceability	Design Traceability Manager
Design Requirements Tradeoffs	Quality Function Deployment Analytical Hierarchy Process
Program Planning/Scheduling	Design Traceability Manager
Mission Profile/Scenario Generation	Mission Decomposition Tool
Mission Decomposition	Concept Mapping (TAKE2) Timeline Management Tool
Functional Flow	TAKE2 Timeline Management Tool
Task Derivation	Timeline Management Tool Sequitur's Workload Analysis System
Action/Information Requirements	Timeline Management Tool
Task/Workload Analysis	Sequitur's Workload Analysis System
Reach, Distance, and Vision Assessment	Computerized Biomechanical Man-model
Geometry, Layout, Structure	Integrated Design Engineering Analysis Software Geometry Interface Tool

Table 2.3.4-3. Developmental Configuration of the CDS Tools (Continued)

FUNCTIONAL REQUIREMENT	CDS TOOL
Control and Display Development	Sherrill-Lubinski's Grapnics Modeling System
Subjective Evaluation of Workload	Subjective Workload Assessment Technique
Flight Test Support	Performance and Workload Evaluation System
Database Management System	INFORMIX
Textual/Graphic Product Development	Microsoft Word MacDraw
Analysis Workstations	v1sg09: Iris 4D/80GT (GMS host) v1sg10: Iris 4D/240GTX (INFORMIX host) v1sg18: Iris Crimson workstation v1sg19: Iris Indigo workstation v1sg20: Iris Onyx workstation Network File System (NFS) and Ethernet
Physiological Data Collection	Workload Assessment Monitor

Table 2.3.4-3. Developmental Configuration of the CDS Tools (Continued)

FUNCTIONAL REQUIREMENT	CDS TOOL
Part-Task Cockpit Prototyping Simulator	Engineering Design Simulator, consisting of: Adjustable cockpit framework Articulating or removable seat McFadden hydraulic control (optional) Cockpit control/display generators: v1sg11: Iris 4D/25TG v1sg12: Iris 4D/25TG v1sg13: Iris 4D/20G v1sg14: Iris 4D/20G v1sg15: Iris 4D/25TG Out-the-window scene generator: v1sg17: Iris 4D/320 Aerodynamic model host: v1sg10: Iris 4D/240GTX Test conductor's console Replaceable stick and throttle Replaceable cockpit side consoles Cockpit display repeaters: Three 6"x6" Sony monitors Two 4"x4" XKD monitors Low-speed data transfer: Ethernet and NFS High-speed data transfer: SCRAMNET network Power/video distribution system Intelligent Input/Output Controller

CSCI #s	Description	Initial # of Items
200 - 299	New Analysis and Design Tool CSCIs	Approx 10
300 - 309	EDSIM Layer 1: Simulation System Software	4
310 - 349	EDSIM Layer 2: Simulation Application Software	Approx 7
350 - 369	EDSIM Layer 3: Cockpit Application Software - CAT Design	10
370 - 399	EDSIM Layer 3: Cockpit Application Software - F-16 Design	12
400 - nnn	EDSIM Layer 3: Future configurations	TBD

Once assigned, the CSCI number will remain fixed for a given CSCI. The version number of the CSCI will be incremented as new versions are released, beginning with version 1.0. The version-numbering scheme is described in the CM Plan (Reference 66, Appendix B). Note that commercial CSCIs will be identified by the version number assigned by the vendor.

Configuration items that have been revised will be identified as Alpha releases of CDS developmental baseline items. This will include the UNIX versions of existing CSCIs and the software items modified by Merit Technology in preparation for Field Demonstration No. 1.

2.3.5 Configuration Management Data Library

The CM data libraries are maintained in Rooms 108 and 109 of Building 248. An extensive audit of the libraries of magnetic media and other hardware items was conducted by the CM Assistant during the reporting period. As a result of this audit, approximately 150 updates and corrections were made to the CM data base.

2.3.6 Change Requests and Change Proposals

Table 2.3.6-1 shows the CPs that were prepared during the reporting period and the CRs that were addressed by each CP. Acronyms used in the tables that follow are defined in the list of Acronyms, Terms, and Abbreviations located in the front matter. The following is the status of the CPs as of June 1993:

Table 2.3.6-1. CRs and CPs Worked During Reporting Period

CP 1: REPLACE CIPLP WITH GEOMETRY INTERFACE TOOL (GIT)

- CR 29: MCAD - INABILITY OF I-DEAS TO WRITE INPUT FILE TO CIPLP
- CR 198: ELIMINATE HARDCODING OF CONTROLS & INDICATORS

CP 2: REPLACE DEN WITH DESIGN TRACEABILITY MANAGER (DTM)

- CR 2: DEN - DOES NOT PROVIDE SUFFICIENT SUPPORT TO PROCESS

CP 3: REPLACE NMT WITH TIMELINE MANAGEMENT TOOL (TMT)

- CR 278: NMT - INADEQUATE SUPPORT OF TIMELINE DECOMPOSITION PROCESS
- CR 114: NMT - ADD DEFAULT MECHANISM FOR COMPUTING START TIMES
- CR 124: NMT - CONTROL CONNECTION-LEVEL RESTRICTION
- CR 125: NMT - IMPROVE TASK CREATION OPTIONS
- CR 126: NMT - REVISE MENU OPERATIONS TO USE P-STRINGS

Table 2.3.6-1. CRs and CPs Worked During Reporting Period (Continued)

- CR 127: NMT - REVISE LIST DATA STRUCTURE
- CR 129: NMT - PROVIDE X-WINDOWS COMPATIBILITY
- CR 130: NMT - RETUNE DYNAMIC MEMORY PACKAGE
- CR 131: NMT - ADD DATA TYPING TO NODE SLOTS
- CR 133: NMT - IMPLEMENT TPEENET TYPE, VERSION, AND SUBTYPE
- CR 134: NMT - SEPARATE DISPLAY PARAMETER SETS
- CR 136: NMT - IMPLEMENT TEST-XXX-STRUCTURE ROUTINES
- CR 137: NMT - IMPROVE TASK CREATION OPTIONS (IDEAL)
- CR 138: NMT - REDESIGN WINDOW INTERFACE LIKE VMS
- CR 139: NMT - DEFAULT VIEW FOCUS
- CR 140: NMT - HELP FEATURE REQUIRES FULL IMPLEMENTATION
- CR 141: NMT - SIMPLIFY TIMELINE UPDATE COMMAND
- CR 142: NMT - REVISE .TNET FILE FORMAT
- CR 143: NMT - MODIFY SEQUENCING OF TIMELINE PROCEDURES
- CR 264: NMT - ADD AUTO FPR GENERATION FROM IATOOL REPORT

CP 4: RESTRUCTURE EDSIM HARDWARE/SOFTWARE ARCHITECTURE

- CR 5: BATS-SP - SEPARATION OF PRIVILEGED & NON-PRIVILEGED ACCOUNTS
- CR 19: BATS-SP - NEED TO IMPLEMENT COMMUNICATION CHECKS
- CR 39: SIMCLP & DISPLAY PROCESSORS - SYSTEM FUNCTIONS NOT SEPARATE FROM COCKPIT DESIGN SOFTWARE
- CR 46: BATS-SP - SHARED MEMORY ADDRESS UNNECESSARY
- CR 60: BSIMUSER - DOES NOT START IN A/A MODE AS STATED
- CR 75: SIMCLP - "INCLUDE" FILES MISSING; SHOULD BE SHARED
- CR 218: BSIMUSER - SHOULD ATTACH TO FDR SERVER DIRECTLY
- CR 220: BSIMUSER - SIMCPL - DEADLOCK WHEN TERMINATED BEFORE SCENARIO RUN
- CR 222: BSIMUSER - FUNCTIONS SHOULD BE COMBINED W/ SIMCLP
- CR 269: BICMIO - RELOCATE DMA, TONE, MCFADDEN BOARDS

CP 5: VMS-TO-UNIX CONVERSION

- CR 275: GENERAL - PORT VMS-BASED CSCIS TO UNIX
- CR 170: GENERAL - INSUFFICIENT NUMBER OF SOFTWARE DEVELOPMENT (Ada, FORTRAN, C) ENVIRONMENTS
- CR 231: DBMS - NEED UNIX-BASED DBMS FOR ANALYSIS TOOLS
- CR 288: X-WINDOWS - PURCHASE OF GRAPHICAL USER INTERFACE BUILDER

CP 6: ACQUIRE ADDITIONAL SGI WORKSTATIONS

CP 7: ESTABLISH QFD DESIGNER AS A CSCI, AS A POSSIBLE REPLACEMENT FOR SUMMET

- CR 235: SUMMET - FAILS TO PROVIDE EFFICIENT ANALYSIS CAPABILITY

CP 8: SUPPORT FOR MERIT TECHNOLOGY PROPRIETARY PRODUCTS

CP 9: ESTABLISH CONCEPT INTERPRETER AS A NEW CSCI

- CR 258: GENERAL - THERE IS A NEED FOR CONCEPT MAPPING CAPABILITIES

Table 2.3.6-1. CRs and CPs Worked During Reporting Period (Continued)

CP 10: OBTAIN SWAS FOR EVALUATION

- CR 204: DCADS - DEFICIENCY IN WORKLOAD PLOT CREATION & INTERPRETATION
- CR 250: FATOOL - CTLA - REPLACEMENT

CP 11: MERIT SUPPORT, FEBRUARY 1993

- CR 22: MDTOOL - DEFAULT VALUE WRITES OVER PREVIOUSLY UPDATED FILES
- CR 28: MDTOOL - OVERWRITING OF FDR FILES
- CR 50: MDTOOL - CRASH CAUSED BY FILE NAME WITH A SPACE IN IT
- CR 96: BATS-SP - USER CAN ENTER TOO MANY VEHEXE PROCESSES
- CR 97: BATS-SP - BATS ONLY ALLOWS 5 AIRCRAFT BUT NO CHECKS ARE MADE
- CR 253: MDTOOL - SYSTEM CRASH WHEN NOT SPECIFYING WIND MODEL
- CR 254: MDTOOL - FORWARD EDGE OF THE BATTLE FIELD (FEBA) IS ABSENT
- CR 255: MDTOOL - FONT SIZE IS ILLEGIBLE
- CR 260: MDTOOL - ABSENCE OF ICONS WHEN REPLAYING FDR FILES

a. CP 1: The status of the GIT development is provided in Section 5.3.

b. CP 2: The status of the DTM development is provided in Section 5.3.

c. CP 3: The status of the TMT development is provided in Section 5.3.

d. CP 4: The status of the restructuring of the EDSIM hardware and software is discussed in Section 5.3.

e. CP 5: The status of the conversion of CSCIs from the VMS to UNIX operating systems is discussed in Section 5.2.

f. CP 6: The recommendation for additional SGI workstations is in work. The CP has not yet been released to the DRB.

g. CP 7: The DRB decided to drop CP7 as a formal CP because QFD Designer is being obtained on a trial basis. A CP will be submitted if and when QFD Designer proves to be a necessary and sufficient component of the CDS.

h. CP 8: The DRB decided that the proposal to obtain Merit Technology support for proprietary products did not require a CP.

i. CP 9: The DRB decided that a formal CP will be required if and when the Concept Interpreter proves to be a necessary and sufficient component of the CDS.

j. CP 10: The DRB decided that a formal CP will be required if and when SWAS proves to be a necessary and sufficient component of the CDS.

k. CP 11: This proposal documents the changes made by Merit Technology personnel during the week of 8-12 February 1993. All work was completed during and immediately after that week.

2.4 Beta Site Management

During this reporting period, the emphasis in Beta Site development was on reviewing requirements to determine what could be accomplished and who could best accomplish it. To that end, a two-stage potential Beta Site process is being developed. The first stage would engage the services of an organization on base that has been involved with CDS development, or that has the capability to host and/or perform several of the components of the CDS. This stage is necessary to allow critical evaluation from outside the development organization. The results will allow the CDS to be fine tuned without exposing an incomplete system for industry participation.

The second stage of Beta Site activity would be the involvement of potential laboratories or groups in the aircraft development industry. This stage will be accomplished to get the CDS into the field of actual cockpit design activity. It may not be feasible to engage an actual cockpit design group working on a production program as a Beta Site. The Independent Research and Development (IRAD) companies or an aircraft contractor working on small scale cockpit research programs may be the best targets as sites. Next, it will be necessary to determine the appropriate timing and the exact target organizations for Beta Site deployment.

Two choices may be available to perform the first stage of the activity. They are the Crew System Ergonomics Information Analysis Center (CSERIAC) and the CSEF.

The CSERIAC was previously involved in reviewing some of the CDS tool updates and replacements, such as MDTOOL, DTM, and TMT. CSERIAC personnel attended demonstrations and provided feedback on these items. Due to the proximity and ongoing involvement, CSERIAC is a good candidate for a first stage Beta Site. Tentative discussions about the possibility were positive.

The CSEF has been closely abreast of most of the activities of the CDS and has the personnel and capabilities to be a good Beta Site for actual performance reviews on cockpits. The next step will be to contact CSEF management to discuss the possibility of becoming a first stage Beta Site.

The second stage Beta Site organizations will likely come from actual aircraft cockpit development companies. The likelihood of obtaining a thorough performance evaluation will be high in these organizations. The process of finalizing subcontracting agreements with four major airframe-manufacturing companies: Lockheed/Fort Worth, Lockheed/Marietta, McDonnell-Douglas, and Northrop, is continuing. These companies participated in the surveys and reviews of the evolving CSDP. This support will facilitate their future involvement in the program. Therefore these companies will be primary candidates for CDS Beta Sites. Discussions about this possibility will be conducted after subcontracting agreements are finalized.

3. NEEDS ASSESSMENT

This section provides information on activities that are being conducted to improve the CSDP and the CDS. It discusses the on-going assessment of both the process and the system, the review of new developmental design tools and methods, and the planned survey of the cockpit design community.

3.1 Assessment of the Crew-Centered System Design Encyclopedia and the Cockpit Design System

This assessment is based on information obtained from the work performed during a trial application of the CSDE and the CDS, and during the program planning and initialization of Field Demonstration No. 1.

The CSDE and the CDS were evaluated to validate capability and sufficiency to help produce cockpit designs in the following ways: (1) through actual performance and analysis of cockpit design activities; (2) on the basis of how well the designer is supported in performing the CSDE activities; and (3) by integration and testing in actual design applications.

Evaluation of the CSDE revealed a lack of depth and direction for a full process description because there were no definitive procedures for each level of cockpit development (Section 4). The CSDE was found to: (1) be too general to be standardized; (2) contain implication of traceability through statements made in the activity summary or product description without providing a medium (computer-based or otherwise) to easily locate a traceable product; (3) have no support for documentation; (4) lack guidance for modeling human capabilities; and (5) only imply iteration of design through discussion but not through the flow of performing cockpit design process activities. In addition, the CSDE addresses full aircraft development programs, making it difficult to apply to a less complex effort such as retrofit or minor avionics modifications. Although designed for flexibility and tailoring, the cockpit analysis activities and tools and the cockpit design activities and tools lack a formal integration process that would promote user understanding. Also, there is a lack of clear and concise guidance for the CDT.

Evaluation of the CDS revealed the need for the development of new software tools and the enhancement of existing ones because many were found to be cumbersome and time consuming to implement (Section 5). In general, many of the tools lack face validity or the capability to correctly predict pilot performance and flexibility. Many of the tools only provide predictions of pilot workload and do not adequately support critical cockpit design processes that must occur during up-front analysis (i.e., deriving mission requirement information), program planning, and the creation of deliverable products (e.g., cockpit design mechanization or traceability documents). Also, maintenance was found to be too costly due to the age of the CDS.

3.2 Survey of Candidates for New Analytical Tools

In an effort to strengthen the CDS and in accordance with Statement of Work (SOW) paragraph 3.4.1.1, the CDS project team reviewed new developmental design tools and methods that were applicable to cockpit design. Increasing emphasis will be given to this requirement in the future, as initial priority was placed on achieving the level of functionality in the present CDS tools to meet the requirements of Field Demonstration No. 1.

During this reporting period, the following periodicals and conferences were used as sources of information for off-the-shelf products considered relevant to the CDS:

a. Proceedings from the STC sponsored jointly by Air Force Systems Command (AFSC) and the Software Technology Support Center (Reference 7);

b. Software Technology Support Center (STSC) software tool evaluation publications, including the *Requirements Analysis and Design Tool Report* (Reference 8);

c. The Falcon bulletin board announcements concerning USAF-sponsored research-and-development programs. (Falcon is an Air Force computer that is used at AL/CFH);

d. Various trade journals, including *Electronic Data News (EDN), Institute of Electric and Electronic Engineers (IEEE) Software,* CSERIAC's *Gateway* newsletter, and *Info World*;

e. Air Force Institute of Technology's (AFIT's) *ComputerSelect* data base on Compact Disk - Read-Only Memory (CD-ROM);

f. The National Aerospace and Electronics Conference (NAECON); and

g. The Professional Development Seminar sponsored by the Association of Computing Machinery (ACM), May 1, 1993, X-Windows applications development tools and techniques.

In addition, the need for possible replacement or augmentation of existing CDS software and hardware tools was identified in thirteen areas. Each area is described in a subsection below.

3.2.1 Tradeoff Analysis

The CSDP calls for a systematic approach to tradeoff analysis in activities associated with up-front analysis, function allocation analysis, and the evaluation of design solution candidates. In the original system, the Survivability Measures and Methods Evaluation Technique (SUMMET) was provided to support these tradeoff analysis activities. The SUMMET software provides decision support for trade-off studies, but contains a number of areas that need improvement in its intrinsic structure, input processes, output processes/uses, output usability; its relationship to other CSCIs; and its relationship to the CDS.

The user must develop the decision structure, and SUMMET does not provide adequate on-line help or guidance for formulating complex decision structures. Similarly, no support is provided to the user for establishing utility functions for decision criteria; therefore, the user is encouraged to use subjective estimates rather than historical or empirical data.

The SUMMET software has no means for reducing subjectivity or bias of user inputs. For example, there is no provision for using a working group approach where team members can supply estimates for their technical area of expertise, or can cross check the input values of others. The user is unable to backup from incorrect menu input selections; the only alternative is to abort the program and lose unsaved, edited data. The program is also sensitive to unexpected inputs and often responds with a stack dump or system crash. In other words, the program attempts to act upon an incorrect input without checking if the input is allowable, such as occurs when text is entered in response to a request for numeric data during the "add utility" and "select type" prompts. Further, when editing a proposal description, there is no indication that a 132-character width is the maximum allowable since the auto wrap does not work. Violating this character-width limit can result in a stack dump and data can be lost. After weighting on a parent node has been performed, the user cannot perform weightings on children nodes with successors. Unless all appropriate nodes have been weighted, the tree will not compress.

The user interface for input processes suffers from two problems. First, completion of some selections from the main menu results in uncontrolled scrolling of information outside the bounds of the viewing window. Second, not all feedback indicators are correct. For example, there may be an indication that certain files have not been created when in fact they have. Although the output from SUMMET assists with trade study decisions by ranking candidate proposals, the complete underlying structure is unknown to the user. For complex models, a graphical presentation of attributes for decision options is required.

Overall usability of SUMMET suffers from several deficiencies. It is possible to intentionally or unintentionally bias results through understanding or ignorance of the software; there are no cross checks (e.g., requests for duplicate entries but in a slightly different format) to suggest that biasing has occurred. The program is difficult to use and requires special knowledge of file creation and retrieval. A significant amount of training and practice are required unless the user has a working knowledge of statistical probability distributions, set up of trade studies, and scaling/interpretation of study output. The documentation is weak and there is a lack of tutorials (or test cases) and online context-sensitive help functions. In summary, the usability does not appear to be well suited to CDS needs. These problems are described in CRs 53, 56, 57, and 235.

The QFD methodology (Reference 2), which is espoused as a structured method for solving problems, is being considered as a possible replacement for SUMMET. Other aircraft design community organizations are developing methods that have evolved from the QFD methodology. Most notably, several contracts in the Avionics Laboratory have dealt specifically with the use of a *house of quality* method for determining which configuration best meets the given requirements.

To supplement the QFD methodology, QFD-related tools are also being evaluated by Veda to foster a systematic approach to the tradeoff analysis activities in the CSDP. Veda's survey of the *ComputerSelect* data base revealed that QFD Designer is the only commercially available tool that implements the QFD process. The QFD Designer contains other tools, such as the AHP (Reference 3), which will also be considered as a possible SUMMET replacement. A copy of QFD Designer was purchased from QualiSoft, Inc., and hosted on an International Business Machines (IBM) PC-compatible workstation. Several CCCD project members attended a one-day workshop on the practices and tools that can be implemented in a tradeoff analysis process. Several separate tools and procedures were presented, including AHP. These tools will be applied and evaluated during Field Demonstration No. 1 for possible inclusion in the CDS.

3.2.2 Subject Matter Expert Knowledge

The CSDP relies on the operational experience and subjective input from subject matter experts (SMEs) of all cockpit areas; however, in the initial configuration of the CDS, there was no support for obtaining and eliciting knowledge from SMEs in a structured, unbiased manner. Also, there was no medium for directly storing information obtained from the SMEs. There was a need to evaluate methods and candidate tools for capturing, structuring, and retaining SME information. Based on current research and development at the Armstrong Laboratory, concept mapping was suggested as a possible solution to this need.

Concept mapping is a knowledge-acquisition technique that has been designed to capture and graphically represent the relationship that exists between concepts in the SME's understanding of the problem space, and the solutions that the expert applies to the problem. Concept mapping is the only known technique that has been proven to be an effective knowledge-acquisition process in several Air Force programs.

The TAKE2 software, an experimental information analysis prototype, supports the creation of concept maps. The TAKE2 software was originally called the Concept Interpreter (Reference 9).

The structure of the concept map can be reformatted and used as input into the relational data base domain. As a data base, the information content of the concept maps can be manipulated and organized to provide further insight into crew station analysis activities.

A copy of TAKE2 was obtained in mid-February 1993 and hosted on a Macintosh computer. Implementing TAKE2 presented very little risk. First, no CDS tool currently depends on the output from this tool, and no CDS tool directly uses its output. Second, TAKE2 was developed for AL/CFH, so the source code ar.d the developer's experience were readily available. Third, the TAKE2 software is Government-owned, and thus required no licensing or maintenance fees. The installation of TAKE2 required no integration effort since it is a standalone application.

3.2.3 Graphical Interface Prototyping

Sherrill-Lubinski's GMS is the CDS tool currently used for the prototyping of graphical PVI devices. GMS suffers from two problems. First, it is difficult to optimize the resulting products to attain real-time performance, as was experienced in implementing the F-16R multifunction displays for Field Demonstration No. 1, because of inefficiencies in the code.

Second, GMS does not provide all of the functions necessary to support control/display prototyping in the CDS context, as illustrated in the following examples:

a. The selection of icons is extremely limited; e.g., when drawing an indicator needle, arrowheads are not available for use. This lack of arrowheads extends the amount of time and effort required to create a display format.

b. A complex fill cannot be done without creating multiple objects. This limitation makes it difficult to implement compound control/display concepts (e.g., putting multiple displays on one panel). Sherrill-Lubinski has acknowledged this limitation as an error but has not indicated when it will be fixed.

c. Differences in the aspect ratios between the workstation display and the cockpit display make it necessary to distort an object to make it appear to be symmetric; (e.g., an object that appears to be a circle on the cockpit display must be defined as an ellipse). This requirement creates problems when items within these nonsymmetric objects (e.g., a needle inside an altimeter) are animated; the needle appears to shrink and grow as it rotates.

d. The GMS preview mode does not redraw occluded objects, so that a moving object permanently erases all objects that it occludes.

e. Finally, GMS does not support the creation of graphical objects directly to physical dimensions. Given the need to design a device to a certain size, a special calibration between the prototyping workstation and the cockpit display must be made.

Other available graphics prototyping system may not solve these problems without introducing other problems. Nonetheless, Veda has been investigating products that might improve the usability and real-time performance of the CDS control/display prototyping system.

Coryphaeus Software, the producers of Designer's Workbench (DWB), provided portions of a Lockheed/Marietta study (Reference 10) in which their product was evaluated against the GMS and Virtual Avionics Prototyping System (VAPS). Coryphaeus furnished the material that pertained to their product, the complete study is not available. The version numbers of the products were also not available. In the excepts provided, it was stated that Lockheed/Marietta found the DWB to be the best overall choice. The excerpt stated that the advantages of the DWB over the

GMS are: (1) the editor is easier to use and (2) the program integration features can be installed more easily between two graphics objects and the program data .

A demonstration version of DWB (Version 2.0) was requested in early April 1993 and provided at Wright-Patterson AFB in early May 1993. Version 2.0 is highly constrained; it includes no developmental tools and there is no way to edit or inspect the displays and the underlying logic, nor to test the program integration features (hooks). Nonetheless, this version was used to drive a Horizontal Situation Indicator (HSI) display, a MultiFunction Display (MFD), and two PFDs. These displays were full-screen displays hosted on the v1sg10 workstation. Performance was found to be about the same as GMS: 9-12 Hertz (Hz) Coryphaeus claimed that there is an error in the demonstration package that slows its performance. In the future, a fully functional version of DWB will be acquired on a temporary loan basis. It will be subjected to a systematic evaluation that will begin with established requirements in the areas of functionality, usability, execution speed, cost (both recurring and non-recurring), and compatibility with other CDS and non-CDS tools. The DWB, VAPS, and GMS will then be evaluated comparatively with the established requirements, and a recommendation (with rationale) will be prepared.

3.2.4 Engineering Design Simulator Aerodynamic Model

Twenty CRS were written regarding the speed, validity, and modifiability of the aerodynamic model in the EDSIM, as documented in Section 3.4.5 of the Delivery Order 9 Final Report (Reference 11). Two possible replacements were investigated for this model to satisfy the requirements for the F-16R project during Field Demonstration No. 1. The only two available, non-proprietary models that were found were the F-16 aerodynamic models that were located at Williams AFB and Edwards AFB.

The Edwards AFB model does not currently run on an SGI workstation and would require a substantial effort to rehost. There is no documentation on the model or its use. Rehosting the model would require extensive work, including the cooperation of Edwards personnel.

The Williams AFB model does run on an SGI workstation, but is tightly linked to other processors in a highly distributed environment. It is written primarily in assembler language, with portions written in the Formula Translator (FORTRAN) and C languages. This would serve to increase the magnitude of the rehosting effort. The Williams AFB model is also unaccompanied by any documentation.

After consulting with Merit Technology to assess the impact of integrating either of these models into the CATBATS, it was agreed that six to eight person-months of labor would be required. Documentation would then have to be prepared. In view of the objectives, priorities, and time constraints of Field Demonstration No. 1, this magnitude of effort could not be justified. The priorities may be adjusted in part as a result of the industry survey that will take place in parallel with the latter phases of Field Demonstration No. 1, in which industry's level of interest in the EDSIM will be gauged. If the simulator is seen as a high priority CDS component, the replacement of the aerodynamic model may be justified. At that point, a further assessment of the capabilities and impact of alternative models will be conducted. In the meantime, the existing model will be used to support Field Demonstration No. 1.

3.2.5 System Logic and Design

One of the areas noted for improvement in the CDS tools is the current lack of support for the system logic and definition process. There are currently no CDS tools that assist in designing and specifying avionics state transitions and logic flows. In the design of controls and displays, it is critical to understand all of the possible variables and the functions associated with the dynamic interactions of cockpit components. With a logic flow tool, a design team would be able to chart each potential function to each subsystem, or to simply trace each path to determine what design integration must take place between subsystems or buses.

In computer science, logic flow tools are part of the Computer-Aided Software Engineering (CASE) environment. More specifically, these tools belong to the family of requirements analysis and high-level design tools referred to as Upper-CASE tools. A comprehensive assessment of Upper-CASE tools was produced by the Air Force STSC (Reference 8). The following tools were identified as candidates for CDS integration: PowerTools, Software Through Pictures (STP), and Statemate In view of current priorities relating to Field Demonstration No. 1, further assessment of tools in this area was postponed.

3.2.6 Cockpit Design System Operating System

To reduce the CDS ownership costs, to improve speed, and to reduce software maintenance costs, the CDS software tools are being converted to a single operating system: UNIX. This permits the elimination of the DEC-proprietary VMS operating system from the architecture, and a migration toward a more efficient, open (non-proprietary) architecture. UNIX is the most widely used operating system and is available on a large variety of platforms, rendering the CDS software more readily portable to other systems. The DoD is encouraging adoption of UNIX-like operating systems. The government's Portable Operating System Interconnect Extension (POSIX) standard is essentially a UNIX standard.

UNIX includes excellent network support. There are two networking products bundled with the UNIX operating system at no additional fee: Transmission Control Protocol/Internet Protocol (TCP/IP) and Network File System (NFS). The TCP/IP, the first of two networking products, is a stream-oriented transport layer protocol and supports a worldwide standard. This protocol guarantees delivery of the data or negative acknowledgment if the transfer fails. The protocol is not hardware-dependent and has been implemented on Ethernet, Pronet, and fiber optic networks. Since Ethernet is provided without extra cost on all UNIX platforms, TCP/IP support is also available without additional software, coding, or cost. In addition, all other non-UNIX platforms have at least one commercial TCP/IP package that allows a basic file transfer, if not higher functions. The most common application programming interface for TCP/IP is Berkeley Sockets This interface allows the stream protocol to look identical to file input/output (I/O), allowing programs that already perform file I/O to perform the same function over the network. These networking capabilities are necessary so that the CDS can be integrated with other manufacturers' platforms and peripherals without additional cost.

The NFS, the second networking product, allows a local file system to be exported and mounted remotely on any number of machines seamlessly and transparently. To the CDS users, the remote system looks like part of their local file systems. This is an excellent method for sharing data files and tools for multiple users because it eliminates the need for a user to move to a different terminal to operate certain CDS tools. The remote mounting feature also reduces the reconfiguration effort when another processor or terminal is added to the EDSIM system.

With UNIX, a program on a given CDS workstation can easily start a program on any other workstation The invoked program's standard terminal interface is routed across the network to the invoking program, and the invoked program cannot tell the difference. This feature is essential to the EDSIM for centrally starting the numerous programs that make up the simulation environment.

This recommendation for the VMS-to-UNIX conversion was documented in CR 275. CDS tools developed under UNIX, and in compliance with X-Windows (Section 3.2.8), are readily transportable to other platforms that also offer these industry-standard systems. The tools can be implemented in any of three languages: C and FORTRAN environments are bundled at no extra cost with the UNIX operating system and an Ada environment was purchased and installed to support the maintenance and development of Ada-based CSCIs.

3.2.7 Data Base Management System

INGRES is a general-purpose DBMS that was delivered as a VMS-based component of the initial CDS configuration. INGRES was found to be inadequate because it was cumbersome, not user-friendly, and quite expensive, not only for the package itself, but also for the support. In the migration to the UNIX platforms, a re-evaluation of INGRES was made. The requirements for a UNIX-based data base package are listed below.

3.2.7.1 Data Base Management System Requirements

a. American National Standards Institute (ANSI) standard Structured Query Language (SQL) compliance, which would allow the easy porting of analysis tools and data bases to other SQL-compliant DBMSs. It also provides a standard interface from which applications can be created.

b. Embedded SQL (ESQL) support to allow data base manipulation from tools that were originally written in a third-generation language such as C, FORTRAN, or Ada. This will be essential in the development of GIT, DTM, and other tool upgrades.

c. Fourth-generation language (4GL) support to quickly manipulate data and generate reports with little code generation.

d. Easy-to-use user interface generation. This includes the possibility to generate a GUI to allow the visualization and manual manipulation of data.

e. Distributed processing capability. Distributed processing power across the new platform's network would significantly reduce network traffic and speed data base access. In this manner, not every user who employs the package from a remote node will have to remotely log into the server.

f. Cost, both non-recurring (purchase price) and recurring (update/maintenance service).

g. Quality of support and maintenance.

3.2.7.2 Data Base Management System Selection

Only three commercial data base packages were available for the UNIX operating system on the SGI workstations, per discussion with SGI marketing personnel, namely: ORACLE, INGRES, and Informix. The following paragraphs compare the three data base package options in terms of the requirements listed above.

a. All three data base packages allow for ANSI-standard SQL data base manipulation.

b. All three packages support ESQL for the SGI platform. However, INGRES provides no ESQL support for Ada, which is a serious deficiency.

c. While some degree of 4GL support is supplied by all three packages, the implementation is specific to the data base and differs in its syntax, versatility, and ease of use.

d. The user/computer interface, except for basic ANSI SQL, varies from DBMS to DBMS. The primary interface factors to consider are forms generation, 4GL, and other features that enhance the interface by making it easier to use. It was apparent from previous experience with INGRES that if a package is hard to use, no one will want to use it. This was a major factor weighing against the reinstatement of INGRES on the new platforms.

e. All three packages support a distributed environment. This, however, becomes costly in terms of software purchase and support. Cost figures were obtained from both Informix and INGRES. These figures showed that the per-user costs of INGRES are higher than Informix, taking into account those options needed to meet the basic requirements of the tools environment. Also considered were those factors that would allow the aforementioned distributed processing. ORACLE has not responded well to inquiries made, but is also historically an expensive package. The per-user costs for Informix are significantly less than those for INGRES. (It should be noted that a *user* constitutes a single application under Windows, so that one person running three DBMS applications would count as three users.)

f. Support and maintenance of the DBMS are critical in the first year of application design and implementation. Fast turnaround to inquiries is essential to keep the project moving forward. It has been our experience thus far that the Informix vendor has been most expedient in responding to inquiries and supplying information. INGRES was slow to respond, while ORACLE said they would send information, but did not do so. Another factor that favored Informix is that SGI is marketing Informix as the preferred DBMS for the IRIS systems, which will go far toward ensuring future supportability.

In view of the above results, Informix was recommended (CP 5) as the replacement for INGRES. Informix Online, ESQL/C, 4GL, and SQL were purchased and installed on v1sg16. To date, Informix has proven [illegible] be an effective means of implementing DBMS-dependent tools and functions, including [illegible] TMT.

3.2.8 Graphical User Interface

A GUI builder was needed to develop windowed Informix applications at an efficient pace. The three leading products were considered: UIM/X, Teleuse, and Builder Xcessory. The criteria for the selection consisted of: (1) the ability to interpret code; (2) a library of well-documented functions; (3) product maturity; (4) ease of learning; (5) ease of use; (6) completeness of documentation; (7) cost; and (8) supportability. Demonstration versions of UIM/X and Builder Xcessory were obtained for evaluation. Teleuse was eliminated early due to its exceptionally high cost, which was more than twice that of the other two systems.

UIM/X (References 12 and 13) received the highest evaluation and was the recommended solution, as documented in the memorandum attachment to CR 288. Two possible sources for UIM/X were identified: SGI and Bluestone Consulting. Bluestone offered a lower price and included a Bluestone Widget Collection at no additional cost.

3.2.9 Workload Analysis

Two CRs (204, 250) cited the need for improvements in the original CDS workload analysis tools. No procedures or users guide was available for the integration of data between the workload-assessment CSCIs. Too many workload calculations were possible and provided voluminous amounts of data that were difficult or could not be integrated and interpreted. Some workload calculations, as well as the tools, lacked validity and predictability. The tools were not sensitive to the small motor movements and cognitive loads imposed by more recent cockpit designs. The recommendation resulting from Veda's work on Delivery Order 10 (Reference 14) was that these tools be replaced by a valid and reliable workload modeling tool.

A set of candidates for tools were evaluated against requirements and on the actual experience of using, or attempting to use, each model. The tool evaluation requirements were as follows: (1) single path versus multipath analysis; (2) program phase applicability; (3) underlying information processing theory; (4) predictive validity; (5) relative complexity; (6) applications; (7) breadth; (8) time requirements for use; (9) interpretation of results ability; (10) associated cost; and (11) cost effectiveness.

The SWAS obtained the highest rating in meeting the CDS needs. The results are presented in Table 3.2.9-1. SWAS predicts channel loading, workload peaks, overall workload, operator requirements, equipment impacts, and time requirements. The data is provided per crewmember task and can be summarized for individual task groupings, partial or full mission segments, or even the entire mission. The SWAS output provides useful analysis information for the workload-assessment activities within the design process. The CSDP identifies those activities in the context of the overall process and relates SWAS to the other CDS tools.

Table 3.2.9-1. Results of Workload Analysis Evaluation

Attributes and Models	Single Path vs. Multi Path	Program Phase Applicability	Underlying Info Processing Theory	Predictive Validity	Relative Complexity	Applications	Breadth	Time Requirements for Use	Interpretation of Results Ability	Associated Cost	Cost Effectiveness
SAINT	Lo	**Hi**	Lo	Lo	Lo	Md	**Hi**	Lo	Lo	Lo	Md
CRAWL	**Hi**	Lo	Lo	Lo	**Hi**	Lo	Lo	Md	Md	Md	Lo
C-TLA	**Hi**	**Hi**	Lo	Lo	Lo	Md	Lo	Lo	Md	Lo	Md
M-SAINT	Lo	**Hi**	Lo	Lo	Lo	Md	**Hi**	Lo	Lo	Lo	Md
SIMWAM	Lo	Md	Lo	Lo	Lo	Md	**Hi**	Lo	Lo	Lo	Lo
WOSTAS	Lo	**Hi**	Lo	Lo	Lo	Md	Md	Lo	Lo	Lo	Lo
HOS	**Hi**	**Hi**	Lo	Lo	Lo	**Hi**	**Hi**	Lo	Lo	Lo	Lo
OWLES	Lo	**Hi**	Lo	Lo	Lo	Md	**Hi**	Lo	Lo	Lo	Lo
SWAS	**Hi**	**Hi**	**Hi**	**Hi**	Md	**Hi**	**Hi**	Md	**Hi**	Md	**Hi**
W/INDEX	**Hi**	Lo	Md	Lo	Lo	Md	**Hi**	**Hi**	Md	**Hi**	Lo
SWAT	**Hi**	Md	Lo	**Hi**	Lo	Lo	Md	Lo	Lo	Lo	Md
TLX	**Hi**	Md	Lo	**Hi**	Lo	Lo	Md	Lo	Lo	Lo	Lo

In February 1993, a copy of SWAS , with license, was obtained from Sequitur Systems on a temporary (18-month) basis. A users manual was included (Reference 5). As a result, an evaluation of SWAS will be made during Field Demonstration No. 1. If SWAS proves to be acceptable, it will be proposed as a permanent member of the CDS tools.

In April 1993, a SWAS trained consultant helped to produce the workload results and provided instruction and training in the application of SWAS to the F-16R project personnel. In the future, the availability of a CDS Users Manual for SWAS will help to control training costs. The temporary nature of the SWAS acquisition will control the magnitude of the initial investment, leaving resources for other alternatives if SWAS is not viable.

3.2.10 Reach and Vision Analysis

An evaluation of the original CDS tool for reach assessment, Operator Assessment of Reach (OAR), revealed several major areas for improvement: (1) the placement of controls can only be defined in a two-dimensional sense, which results in all controls being internally represented as flat-panel surfaces; (2) no capability is provided to identify obstructions to portions of the limbs and joints; (3) there are no clothing options available in the anthropomorphic data, and no capability to include chemical/biological protection equipment; and (4) the user interface is inadequate in both input and output phases.

The evaluation of the CDS vision assessment tool, External Vision Analysis Program (E-VISION), revealed additional areas for improvement: (1) there is no capability to assess eye protection requirements; (2) data input procedures are tedious; and (3) there is no capability to overlay Military Standard (MIL-STD)-1850B requirements.

Two candidates were surveyed for replacement of OAR and E-VISION: Computerized Biomechanical Man-Model (COMBIMAN) and JACK. Copies of the JACK Users Guide and Programmers Guide were obtained. The evaluation indicated that this tool is not compatible with I-DEAS or any off-the-shelf Computer Aided Design/Computer Aided Manufacturing (CAD/CAM) system because JACK uses its own unique system. Additionally, validation data could not be obtained on the tool.

The CDS Upgrade Plan (Reference 15) recommends the installation of COMBIMAN as a replacement for OAR and E-VISION. COMBIMAN has the capabilities to satisfy the above-mentioned needs. COMBIMAN was validated by the USAF and used in analysis activities by several DoD aircraft production programs. Assistance and documentation are readily available because it is an Armstrong Laboratory product. The COMBIMAN support contractor has begun to rehost the tool on the appropriate CDS SGI workstation.

3.2.11 F-16 Throttle and Sidestick

To configure the EDSIM for Field Demonstration No. 1, an F-16-like sidestick, an F-16 Block 30/40-like throttle, and a patch junction box with power were required. Veda identified several alternative solutions through discussions with knowledgeable personnel from the Armstrong and Wright Laboratories.

The most cost-effective source was Technical Products, Inc. (TPI). The TPI simulator products are very affordable and have been successfully used in several local laboratory facilities. The F-16 sidestick was received and installed in late March 1993. TPI modified their F-16 throttle to make it a more realistic representation of the actual device and it was installed in July 1993. TPI also designed and delivered panels that facilitate installation and removal of the control devices.

3.2.12 Design Traceability

As specified in the System Segment Specification (Reference 16), the Designer's Electronic Notebook (DEN, Reference 21) was intended to guide the cockpit designer through the CSDP, displaying the process to the designer and allowing him/her to step through its various phases and activities. It was also intended to allow the designer to launch various CDS tools, to have access to a daily electronic logbook, a lessons-learned data base, and a glossary of terms used in the design project. However, the DEN that was delivered with the CDS was known to be deficient in satisfying any of the requirements stated above. This fact was determined through an assessment made during the Delivery Order 10 contract (Reference 14). Through the process described below, Veda determined that the best solution was to develop a totally new tool called the DTM to replace the DEN.

Veda prepared a more detailed set of requirements and documented them in the DTM Design Document (Reference 17). A DTM Verification Test Plan was delivered as an attachment to the DTM Design Document.

A survey was conducted of off-the-shelf products to determine their ability to satisfy the DEN requirements. SDRC's Data Management Control System (DMCS) and Protocol's Requirements Traceability Tool (RTrace) were carefully considered, but these components did not provide the necessary functionality and had extremely costly license and maintenance fees. Other alternatives, such as the DEC Code Management System (CMS) and Module Management System (MMS), and the UNIX Revision Control System (RCS) provided small subsets of the required functionality but could not be integrated to provide the necessary support.

The final conclusion of our make-or-buy analysis is documented in CP 2. DTM is being implemented in Informix and runs on the CDS workstations. When fully complete, the DTM will provide the following capabilities: (1) access to all other UNIX-based CDS tools; (2) a user-friendly, windowed interface; (3) record keeping for day-to-day activities; (4) file support for multiple users working multiple projects, storing and retrieving data for the selected project and user; (5) guidance in the CSDP and CDS tool usage; (6) design and decision traceability; (7) management support for each project (scheduling, statusing, reporting); (8) the means to update the procedures.

Section 5.3.2 of this report provides further information on the implementation of the DTM.

3.2.13 Mission Description and Decomposition

The CDS tools includes Merit Technology's MDTOOL (Reference 4) as a means of generating mission descriptions and mission timelines. While MDTOOL is fully functional, it has some deficiencies that might be satisfied by an off-the-shelf alternative. These deficiencies have been documented in several Change Requests. One general deficiency is that MDTOOL is weak in its support of air-to-air mission planning. Also, its user interface is not intuitive to the operations analyst; effi·ent use requires in-depth training. The interface does not furnish a desirable level of error-hand and user feedback. Due to these and other needed improvements, possible replaceme for MDTOOL are being researched.

Du ng the reporting period, two alternatives were evaluated: the Tactical Aircraft Mission Planning System (TAMPS) and the Air Force Mission Support System (AFMSS). On 29 October 1992, Veda and CCCD Program Office personnel visited the McDonnell Douglas Corporation in St. Louis to attend a demonstration of TAMPS. The TAMPS software uses various types of Defense Mapping Agency (DMA) data to support mission planning: Digital Terrain Elevation Data (DTED); imagery from satellite photos; Digital Chart Data, and World Vector Shoreline.

TAMPS uses aircraft performance data from twenty-six lookup tables. Other TAMPS capabilities include weapons and stores loadout data; target maneuver calculations such as laydown, pop-up, straight-path dive; and loft maneuvers; weapon delivery calculations; threat avoidance displays; mission analysis; and production of a combat mission folder.

The TAMPS software was then evaluated in light of CDS requirements and documented in the trip report (Reference 18). It was concluded that TAMPS offers capabilities similar to MDTOOL in many areas, and substantially better functionality in several other areas. There are four major drawbacks:

a. The TAMPS software requires modification to produce the Event Timeline (ETL) and the EDSIM interface files;

b. The TAMPS software offers no support for the generation of air-to-air scenarios;

c. The TAMPS host (a Sun SparcStation) is not available in the current CDS architecture;

d. The TAMPS interface is poor. McDonnell Douglas is currently redesigning it in Motif.

It was recommended that the Air Force request a no-cost copy of the TAMPS source code to investigate the cost of developing the necessary upgrades. It could be hosted on a temporary basis on a Sun SparcStation in the off-site Veda facility for hands-on evaluation. No further action has yet been taken on this recommendation.

The AFMSS contract was recently awarded to Lockheed/Sanders Division. The AFMSS will become a widespread Air Force standard, and thus would be a valuable addition to the CDS. It will replace the Mission Support System (MSS) II, which some USAF agencies are now using. Veda received a demonstration of the AFMSS user interface at the 1993 NAECON convention. It appeared to be a highly effective interface. The AFMSS status and objectives were then discussed with the government's technical representative. In view of the current developmental status of the AFMSS, it would be premature to recommend inclusion in the CDS at this time; however, Veda will continue to monitor its development for possible integration into the CDS tools at a future date.

3.3 Cockpit Design Community Survey

Survey materials are being compiled for distribution to the cockpit design community to obtain information regarding future requirements and implementation of the CSDP and the CDS. The package will be sent to qualified reviewers for input and comments under a subcontract agreement. Information from these reviews will be consolidated and, along with lessons learned during Field Demonstration No. 1, will form the basis for the CDS upgrade planning.

3.3.1 Need for Survey

The rationale behind the Industry and Government survey is to take into account the end user requirements for the CSDP and the CDS. An end user is defined as any cockpit design organization that has a direct impact on the outcome of aircraft cockpit designs. Two kinds of organizations have a direct effect on cockpit designs: an aircraft manufacturer cockpit design group, and a government System Program Office (SPO) cockpit design group. The aircraft cockpit group directly performs the analyses, design, and evaluation of the cockpit system, while the SPO group helps to structure the aircraft user (command-specific) requirements and ensures that the cockpit meets or exceeds those specific requirements. Both are responsible to higher organizations and require specific products from their activities. The focus is to provide a system that can perform all necessary activities in a timely manner to meet or exceed the needs of the crew and the mission primarily and other systems secondarily.

3.3.2 Survey Development

To gain a perspective from the cockpit design community, a medium to gather information was required; therefore, a comprehensive survey package was designed. This package includes a narrative description of the CSDP that illustrates its design process activities, inspectable products, and computer support tools. A network model of the process was constructed to elicit industry feedback. The network contains activity nodes that describe tasks throughout a typical design problem, and product nodes that outline the content of intermediate and final products such as plans and reports. The graphical representation of the network was developed in MacDraw Pro, and the textual portions were developed in Microsoft Word.

The cockpit design activities were identified and sequenced to a level of detail at which individual tasks, and the products they produce, could be seen. Each series of activities was logically aggregated into products such as specifications, cockpit design documents, and traceability documents for major-milestone activities. Once this top-level breakout was accomplished, it was concluded that industry feedback (regarding their cockpit design process) was necessary before continuing with the full description of the process.

3.3.3 Survey Content

The industry/government survey package consists of instructions to the evaluator, a full-color graphical and textual description of the process (which include procedures, data bases and tool references) with a questionnaire, a scenario walk-through of the process as it would be applied to a hypothetical problem (with embedded questions), and a post-evaluation questionnaire. Some specific questions are asked in the scenario, and written recommendations for potential CSDP and CDS requirements and design priorities are requested. Also, site visits with each of the evaluators are planned to ensure that information pertinent to the process and the design support tools has been captured.

The industry/government survey package solicits feedback concerning the viability, acceptance, and validity of the CSDP and the CDS. Recommendations will be sought regarding fine details needed in the CSDP and on whether the CDS tools are considered useful. Information will be requested through both written reviews and subsequent formal face-to-face discussions. Together, the results from these inquiries will be used to help define requirements for future upgrades to the CSDP and the CDS.

3.3.3.1 Evaluator Instructions

To prepare those in industry and government program offices for evaluation of the CSDP and the CDS, an introductory letter and a pre-evaluation instructions were provided to precede the descriptive process overview and scenario walk-through of the process.

3.3.3.2 Crew-Centered System Design Process Description

The CSDP description contains a top-level view of how to use the CDS (along with its procedures, data bases and tools) to perform a time-critical series of design iterations. The structure of the process activities is broken down into five major classes of activities: Up-Front Analyses, Program Planning, Crew System Analyses, Crew System Design, and Crew System Evaluation. Each area is dependent on the other and employs multiple iterations to develop and refine the cockpit design.

3.3.3.3 Scenario Walk-Through

The scenario walk-through is arranged in nine segments that describe a hypothetical cockpit problem, the CSDP, the work environment, mission decomposition, traceability, design activity, testing, CDS-generated products, and a conclusion. At the end of each segment, the reviewer is asked to judge how well the characters in the scenario are able to use the features and capabilities of the CSDP and the CDS to solve some aspect of the cockpit design problem. These questions elicit opinion about the benefits and costs of the CSDP and the CDS, the credibility of the underlying technology, the perceived organizational changes when using the CSDP and the CDS, and the satisfaction of the cockpit solution. The answers to the questions will provide a means to establish and rank present and future system requirements.

A scenario presented in this way will complement the CSDP description. Whereas the process describes how the technical content of the CDS is intended to support the CSDP, the scenario measures user sentiment about the acceptability, validity, and viability of both the process and the tools. Information from both sources should be useful in determining requirement trade-offs. CSDP development information can be found in Section 4 of this report. A current working version of the CSDP description is provided in Appendix C (Reference 66). The scenario walk-through is provided in Appendix E (Reference 66).

3.3.3.4 Post-Evaluation Questionnaire

The final component of the industry/government review package is the post-evaluation questionnaire. The questionnaire probes the composition of a design team, the work environment (to include computer hardware and software), and the reaction of the evaluator to various design objectives.

3.3.3.5 Follow-Up Review Sessions

Once the survey material has been delivered, a period of three weeks will be allowed for the review teams to perform the evaluation of the process and to answer all of the questions. Clarification questions from the reviewers have been encouraged and will be answered by telephone prior to the face-to-face meeting. The reviewers will then return the materials and Veda will examine the replies and formulate a discussion briefing that will be presented to each specific review team. Soon afterward, a face-to-face discussion will be conducted with each review team concerning the information that each has provided. These discussions are vital to the success of the program and will ensure that the feedback is interpreted properly. Approximately one day will be spent with each team to discuss conclusions and to ask supplementary questions. This initial interchange is considered a critical step in the quest to enhance the CSDP and in the establishment of an open discussion link with the review teams.

4. CREW-CENTERED SYSTEM DESIGN PROCESS DEVELOPMENT

The fundamental element of crew-centered cockpit design is the effectiveness of the CSDP. The CSDP is intended to provide a CDT with information and guidance that will enable the development of quality cockpit designs in a timely manner. The thrust of the CSDP development is to build a process that the aircraft industry will *want* to use, and to provide the supporting procedures and tools that will allow it to be applied correctly and efficiently. To do this, an assessment is being made of the CSDP as well as typical, current, and past industry practices. This assessment is being done to allow for possible improvements to the CSDP and to industry practices that will better support the production of crew-centered cockpit designs on a consistent basis.

The intent is to compile a workable set of activities that are necessary and sufficient components of the cockpit design process, using both interactive and iterative attributes since they are critical aspects of that process. To define, create, and evaluate a cockpit depends on the ability of the CDT to: (1) apply the results of one activity as the input for the next related activity (interaction), and (2) determine when and how to perform the task of re-assessing, re-designing, and re-evaluating (iteration). The number of interactions depends on the number of baseline activities chosen, whereas the number of iterations depends on the quality desired and the time allowed for development.

4.1 Process Review

Two separate CSDP activity assessments were made since the initial delivery of the CSDP. The first assessment was performed prior to this contract as a part of Delivery Order 10 (Reference 14) and the second assessment was performed during the first quarter of this contract. As a result of the assessments, it was discovered that the original CSDP was more like an encyclopedia than a process. Subsequently, the term "Crew-Centered System Design Encyclopedia" (CSDE) came to represent the original CSDP.

In both assessments, a lack of depth was found in the CSDE lowest level pages that contain specific procedures and other planning activities. Specifically, there were no real procedures or direction for a process (definitive step-by-step instructions). An immense amount of information existed regarding every potential procedure that could affect the outcome of a cockpit design; however, there were no practical procedures for performing program planning or cockpit design. In addition to the lack of depth, no meaningful process description for cockpit development was available. Only a minimum of information was found regarding: (1) what to specifically perform; (2) how to interact with other tasks; and (3) how to iteratively evolve the design. Additionally, the ability to follow the CSDE procedures for each activity was hampered by the fact that only a top-level description of how to complete each procedure was provided.

In addition, the CSDE did not attempt to explain the use of either interaction or iteration; while the CSDP attempts to define usage by placing procedures in each activity to explain how to use results obtained from other activities. The CSDP also attempts to explain how to use the cycle of analysis, design, and evaluation as built-in activities on any program.

4.2 Process Design

With the discovery of the lack of guidance for actual use of the CSDE, it was necessary to put together a *strawman* process that reflects the dynamics and interdependencies of real-world cockpit design. For this representative process, MIL-STD-46855 and its associated Data Item Descriptions (DIDs) were used for guidance. Additionally, experience with many types of aircraft designs, most prominently the F-22, B-2, and C-17 cockpits, was used to provide insight into the formal military standard implementations.

Since there is more than one way to design the CSDP, new methods are being investigated on a continuing basis for improving the CSDP, delineating its activities, and providing more systematic tools to capture quantitative results. Specifically germane to all forms of improvement are the key abilities to produce fully recognizable products, and to prepare those products in a timely fashion that will not impact the development of other systems on-board the aircraft. The CSDP will continue to evolve as the efforts described in the following sections are completed.

4.3 Process Description

A description of the CSDP (also called the CCCD Process) was written in order to begin implementation of the computer software within the CDS (it does not include programming details) and to gain feedback on its development. This description (Reference 66, Appendix C) represents requirements that currently are not implemented for Field Demonstration No. 1. Enhancements will be made based on the feedback received from government and industry.

The DTM (Section 5.3.2) now provides access to the CSDP activities at appropriate times. The CSDE will continue to be updated over the life of the contract because many of the original activities that are documented in it are valid; however, they must be finalized and/or explained at a finer level of detail so that they will be effective. The CSDE will become a data base to hold procedures that can be added to or deleted from the CSDP according to the need of a particular program.

4.4 Process Implementation Activities

As many facets of the CSDP as feasible were implemented for Field Demonstration No. 1. A decision was made to focus on pre-established program goals, or *marks of progress,* defined for Field Demonstration No. 1 and described in the Validation Test Plan (Reference 19). The follow-on will be as full an implementation of activities as practical by Field Demonstration No. 2, with completion by Field Demonstration No. 3. Through continued application and feedback from source experts, the CSDP requirements can be refined. Iterative and interactive development are the two most critical attributes of cockpit design and should therefore contribute the most towards the full development of the CSDP.

During Field Demonstration No. 1, CSDP activities are performed using the new procedures, along with the current set of upgraded design tools (i.e., SWAS). The area of Crew System Analysis was determined to be the best defined (in terms of detailed procedures) for verification and recommendation of upgrades to the process and tools. Activities were defined for the entire process, while step-by-step procedures to perform those activities were written for only certain Crew System Analysis portions of the CSDP. Specifics of each of these areas will be reflected in the results of Field Demonstration No. 1, which will be presented during a Bi-Monthly Progress Review. The written record will be placed on the DAL.

4.5 Process Application

Due to parallel development, it was known that some procedures and tools required to support activities would not always be available for application in Field Demonstration No. 1. In order to perform various parts of Field Demonstration No. 1 in a more effective manner, information pages containing specific procedures and other planning information were developed for selected critical activities. The information pages provide the following detailed information fields: *Activity Definition, Preceding Activity, Succeeding Activity, Procedures, Recommended Software Tool, and Recommended Data Bases.* These fields were defined to Perform Mission Profile Analysis, Perform Mission Scenario Analysis, Perform Functional Flow Analysis, Perform Action/Information Analysis, and Perform Task Workload Analysis, all of which are CSDP activities that are being exercised during Field Demonstration No. 1. The full documentation that shows how the CSDP contributed to accomplishing cockpit design will be available after the completion of Field Demonstration No. 1. This documentation will be listed in the DAL and will be available in the Validation Test Plan results. These results will contain an assessment of both the fully developed critical activities and those potential activities defined after the field demonstration.

4.6 Process Evolution

The CSDP technical description (Reference 66, Appendix C), an example scenario walk-through (Reference 66, Appendix E), and user and evaluator questionnaires (Reference 66, Appendices D and F), were prepared for industry and government review. Comments from the Industry/Government Review and the Veda Team Review will be assessed. The intent is to analyze the findings and update the CSDP technical description to reflect the needs of the end users. All suggestions will be taken into consideration to determine if they improve the quality of the process. Quality and traceability to crew and mission requirements will be the guiding factors in the evaluation process. The results from Field Demonstration No. 1 and the Industry/Government Review will determine the requirements for subsequent CSDP enhancements and field demonstrations.

5. COCKPIT DESIGN SYSTEM UPGRADE IMPLEMENTATION

A number of the CDS commercial hardware and software components are more than seven years old; few components are newer than three years old. Every two to three years, the computer industry reduces processor costs by a factor of two, while processing capabilities improve by a higher factor. New versions of operating systems and software tools are constantly improving and must be upgraded. Therefore, an upgrade of the CDS commercial hardware and software components was needed to achieve performance improvements, with attendant reductions in purchase costs, maintenance costs, floor space requirements, and cooling needs. Since an unlimited number of upgrade paths were possible, it was imperative that the search for the best approach remain focused on the goals and requirements of the CSDP. This initial upgrade will be followed by a major upgrade to the CDS (adding or replacing applications software) later in the contract.

This section discusses the initial system upgrade requirements and the subsequent tool development that was necessary to accomplish the improvements.

5.1 Initial System Upgrade

a. **Background.** This system upgrade was driven by the need to reduce the ownership cost of the CDS and to improve its performance, in addition to the following requirements:

(1) To support field demonstration mission analysis software requirements such as a limited number and type of ground-based threats; the Persian Gulf geographical setting and required Digital Feature Analysis Data (DFAD) features; the types of weapons and sensors to be used; the types of PVI devices to be employed; and the data to be gathered and analyzed. The Advanced Tactical Air Reconnaissance System (ATARS) sensor required an additional workstation to enable the sensor view and an additional workstation to handle the processing required to implement the mechanization. Section 6.1.3.1 provides further information on the scenario requirements.

(2) To meet the needs of the aircraft industry, potential Beta Sites, and the cockpit design and evaluation community. These needs deal with the nature and scope of the simulated environment such as the PVI, and the host hardware and software systems that are currently in use and/or are planned for use.

The Delivery Order 9 Upgrade Plan (Reference 15), discusses a number of recommended changes to the basic CDS. Each recommended change includes the rationale, implementation considerations, and cost. The recommendations in the Upgrade Plan (Sections 5.1 through 5.7 therein) represent a baseline around which an integrated design is being developed. Implementation of the majority of the recommendations was accomplished. The software structure of the CDS is sound and only small changes to the basic structure are anticipated unless there is a substantial reason as determined by the data generated from the industry visits.

b. **Progress.** Two SGI workstations are being purchased and will be used to conduct real-time simulation trials during field demonstrations (Change Proposal 6). The first workstation, the Onyx v1sg20, will be used as the CATBATS host (currently hosted on v1sg10) and will offer the following capabilities:

(1) Software compatibility with the existing UNIX/SGI environment;

(2) Compatibility with the existing Ethernet network;

(3) Flexibility and expandability to accommodate future CDS needs and to promote ease of growth;

(4) At least four 100-megahertz (MHz) processors;

(5) At least 128 MB of memory;

(6) At least a 25% increase in the CATBATS update rate.

Figure 5.1-1 shows the current SGI configuration as of June 1993.

The Onyx workstation was the recommended choice as a real-time processor. The initial configuration of four R4400 processors, each of which runs at 100 MHz, will meet the computational requirements listed above and may be configured and expanded to accommodate up to 24 processors. The Onyx is expected to improve CATBATS at the rate of four times its existing computation speed.

The following information outlines the specifications for the Onyx workstation:

(1) Onyx deskside 4-Central Processing Unit (CPU) 100 MHz workstation;

(2) 64 MB memory;

(3) 1.2-gigabyte disk;

(4) Iris development option for IRIX 5.0 only;

(5) NFS;

(6) Full extended warranty.

The second workstation, Crimson v1sg18, will be used for analysis and development work. Requirements for the second workstation were the same as for the first workstation with two exceptions: (1) the memory requirements were reduced to 64 MB, and (2) there is no need for multiple processors.

The following information outlines the specifications for the Crimson workstation:

(1) Crimson/Reality Engine Graphics Supercomputer with 64 MB of memory;

(2) 1.2-gigabyte Small Computer System Interface (SCSI) disk;

(3) CD-ROM drive;

(4) Iris development option;

(5) NFS;

(6) Iris performer visual simulation software;

(7) Full extended warranty;

(8) One 100-Mhz processor.

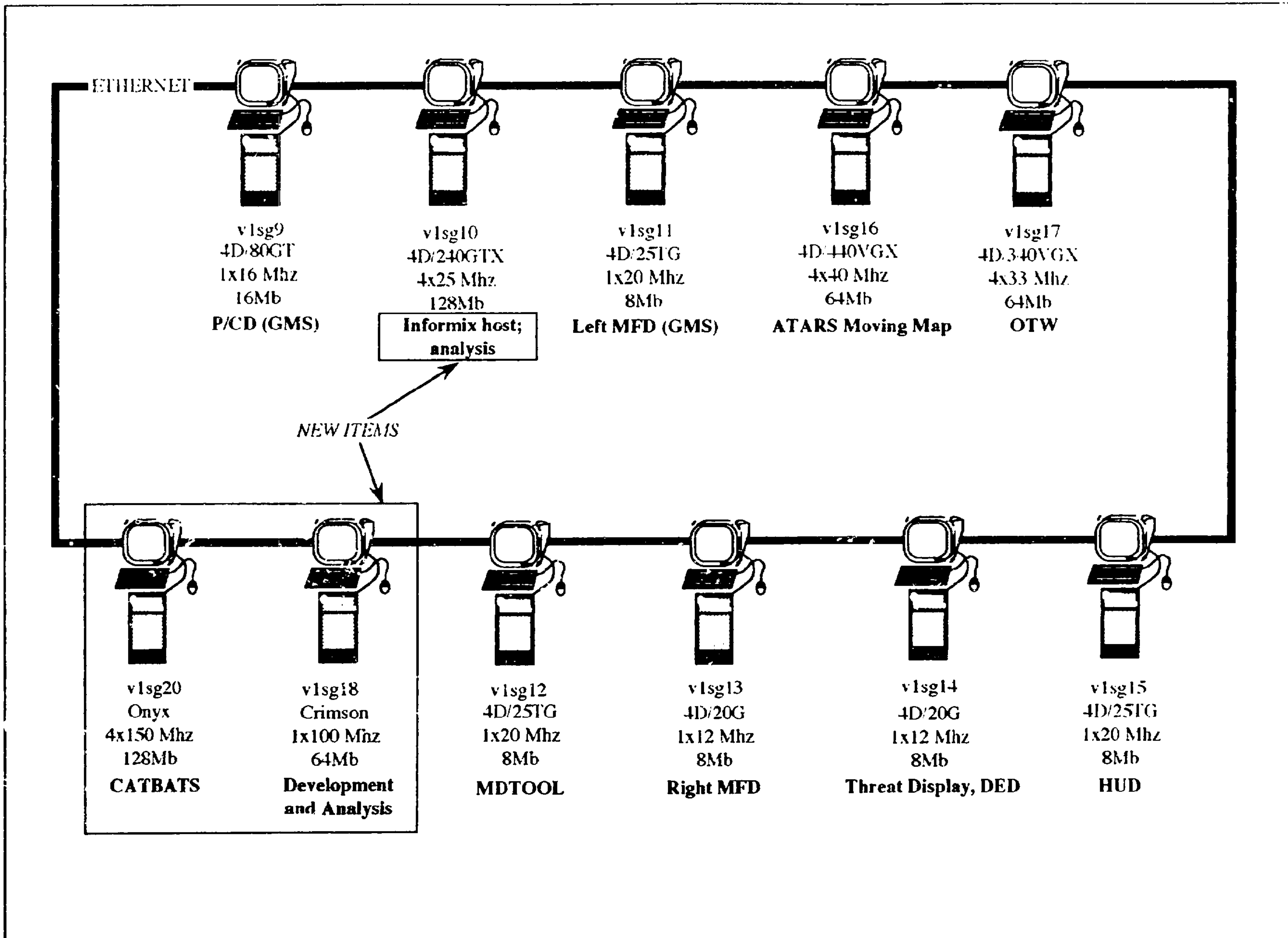


Figure 5.1-1. SGI Configuration as of June 1993

Both the Onyx and Crimson workstations are fully compatible with existing SGI workstations. Code may be ported without recompilation. The ethernet interfaces are compatible with the present SGI configuration and will require no hardware or software changes.

A color printer, digitizing table, and a high-speed scanner are currently under analysis for further enhancements to the CDS.

5.2 VMS-to-UNIX Integration and Conversion

This section contains the technical details of the CSCIs modified to convert from the VAX-hosted VMS operating system to the SGI-hosted UNIX operating system. It presents information on how each CSCI was affected, including porting difficulties, and the current porting status of each.

a. **Background**. The VMS-to-UNIX conversion priorities were originally stated in the Delivery Order 9 Upgrade Plan (Reference 15), submitted 15 January 1993, and were modified in CP 5, submitted 17 February 1993. The converted CSCIs are those that support Crew System Analysis activities during cockpit design: Procedure-Level Timeline Analysis, Task-Level Timeline Analysis, and Geometry Analysis. Each conversion was completed in four steps by: (1) porting the code from the VAX to the SGI via TCP/IP; (2) compiling the code on the SGI and correcting compilation errors as they occurred; (3) verifying the code by running it and using input data files generated during the Delivery Order 10 trial application and then comparing the output to the output obtained from the original VAX version; and (4) modifying the documentation as needed to reflect changes.

Tables 5.2.1-1 and 5.2.1-2 are replicas of the original tables submitted in the Upgrade Plan. Table 5.2.1-1 shows each CSCI number and name in priority order (greatest priority assigned to those CSCIs that directly support the crew system analysis activities) for porting, along with CSCIs that are to be removed or replaced, and gives the current status that includes: Comp (conversion or replacement is complete); Plan (conversion or replacement is being planned or conversion or replacement has not yet started); Work (conversion or replacement is in-process). No priorities were assigned to those CSCIs that are to be replaced by UNIX equivalents or off-the-shelf commercial products. Although a priority was assigned to each of the other CSCIs, completion was not necessarily accomplished in the assigned order. The conversions were done in parallel and, because the amount of time required for conversion varied widely, several of the lower-priority conversions were completed before some of the higher-priority items.

(1) 57 - DEC PostScript laser printer driver (POSTDRV). This CSCI became obsolete due to its dependency on the VAX/VMS architecture. The UNIX architecture maintains its own printer drivers. A laser printer will be obtained to test the printing compatibility of the SGI drivers and printer, and an assessment will then be made to determine if the SGI drivers require upgrading or if the printing capability is sufficient. Status: Awaiting delivery of a laser printer from the CCCD Program Office to test the printing capabilities of the SGI.

(2) 23 - Survivability Measures Methods Evaluation Technique (SUMMET). CR 235 stated that SUMMET failed to provide sufficient analysis capability. In response to this CR, a candidate for tool replacement was assessed. The QFD Designer was evaluated after training sessions were attended. A determination was made that the QFD Designer provided more functionality than is required for trade-study support and it was proposed that the AHP methodology, a subset of the QFD methodology, may be more appropriate. After a full assessment of the AHP tool (Expert Choice), a formal CP will be written to recommend an appropriate replacement for SUMMET in response to CR 235. Status: Awaiting a final assessment of the QFD Designer that will be accomplished at the end of Field Demonstration No. 1.

Table 5.2.1-1. Critical CSCIs Affected by VMS-to-UNIX Conversion

Critical Priority	CSCI Name	Port	Replace	Remove	Status
*	10 ADA Ada development system		UNIX equivalent		Comp
*	11 C C development system		UNIX equivalent		Comp
*	13 F77 FORTRAN development system		UNIX equivalent		Comp
*	48 TCP/IP Network interface package		UNIX equivalent		Comp
*	55 VWS DEC Windowing System		X-Windows		Comp
*	57 POSTDRV DEC PostScript laser printer driver		•		Plan
*	104 PRSCRN PRint SCReeN utility		•		Comp
1	3 DEN Designers Electronic Notebook		Design Traceability Manager		Work
2	5 DBMS Data Base Management System		Informix		Comp
3	6 NMT Network Management Tool		Timeline Management Tool		Work
4	20 I-DEAS Mechanical Computer Aided Drawing		UNIX equivalent		Comp
5	102 CIPLP Cockpit Instrument Layout Program		Geometry Interface Tool		Work
6	24 IATOOL Information Analysis Tool		Timeline Management Tool		Work
7	76 IADBX Data file to support IATOOL		Timeline Management Tool		Work
8	33 CCC Cockpit Configuration Control	•			Comp
9	32 FATOOL Functional Analysis Tool		Design Traceability Manager		Work
10	163 CSAT Crew System Analysis Tools		Design Traceability Manager		Work
11	50 MSA Mission Scenario Analysis	•			Comp
12	116 MPE Mission Procedure Evaluation	•			Comp
13	51 MTA Mission Timeline Analysis	•			Comp
14	106 MTP Mission Task-time Probability	•			Comp
15	101 PLTGGP MTA & MTP workload plot program	•			Work
16	7 DI-3000 Specialized graphics library		UNIX equivalent		Comp
17	113 PLOT3D3 2D & 3D graphics plot program	•			Comp
18	23 SUMMIT Trade-off Study Tool		•		Plan
19	153 C-TLAPREP Timeline Analysis Preparation	•	Design Traceability Manager		Plan
20	154 MSA-CTET Convert Mission Scenario File	•			Work
21	155 C-TET Task Execution Time	•			Work
22	156 CTET-CTLA Convert Mission Scenario File	•			Work
23	158 C-TLA Timeline Analysis	•			Comp
24	159 C-TLAT Timeline Analysis Translator	•			Comp
25	160 C-TLAP Timeline Plot	•			Work
26	161 C-WLH3D Workload 3D Histogram Generator	•			Work
27	30 DLA Display Legibility Analysis	•			Comp
28	21 OAR Operator Assessment of Reach		Cmptr Bio-Mech Man-model		Comp
29	117 MCOS Monte Carlo Operator Sample Generator	•			Work
30	27 E-VISION External Vision Model		Cmptr Bio-Mech Man-model		Comp

Table 5.2.1-2. Non-Critical CSCIs Affected by VMS-to-UNIX Conversion

CSCI Name	Port	Replace	Remove	Status
9 MASS-11 Word Processor		Word for Macintosh		Comp
12 PASCAL PASCAL development system			•	Comp
22 C-SAINT CAT Systems Analysis of Tasks		Sequitur Wkld Anl SW		Comp
28 W/INDEX Workload Index	•	Sequitur Wkld Anl SW		Comp
29 SWAT Subjective Workload Assessment Technique	•	PC version		Plan
31 PET Procedure Execution Time	•			Comp
34 AED Automated Experimental Design			•	Comp
41 COLOR Display Modeling and Algorithmic Color			•	Comp
42 SPREADS Access Technology spreadsheet program			•	Comp
43 TPS Interleaf Technical Publishing Software			•	Comp
49 HANDLE Defense Mapping Agency data processor		CD version		Comp
60 CSDSDB Crew System Design & Spec Data Base		•		Plan
62 ACPCDB Aircraft System and Crew Perform DB		•		Plan
64 CSPDB Research Data Crew System Performance		•		Plan
65 PPWEDB Test Data on Pilot Perf, Workld, & Error		•		Plan
67 LLDB Lessons Learned Data Base		•		Plan
70 MSADB Abstracts of Military Stds Data Base		•		Plan
72 TECHDB Technology Assessments Data Base		•		Plan
105 CSTPO C-Saint Preparation Zero Program		Sequitur Wkld Anl SW		Comp
108 C-TAC CAT customized Tac Brawler	•			Plan
110 RVRA Runway Visual Range Analysis	•			Plan
112 FAR Function Analysis Report	•			Work
124 CSTGRF C-SAINT Graphical post processor		Sequitur Wkld Anl SW		Comp
125 CSTSUM C-SAINT SUMmary post processor		Sequitur Wkld Anl SW		Comp
126 AASPEM-CAT interface AASPEM & MDTOOL	•			Plan
127 TBMAIN Tac Brawler Main Program	•			Plan
128 SUMAIN Tac Brawler Summary program	•			Plan
129 IFACE Tac Brawler Interface program	•			Plan
132 REFORMAT Reformat program, supports PRSCRN			•	Comp
133 FIXUISPDES Bug fix for VMS Print Software			•	Comp
138 DECCMS DEC Code Management System			•	Comp
139 DECMMS DEC Module Management System			•	Comp
141 LSE Language Sensitive Editor			•	Comp
145 TEAMAIN Terrain Encounter Analysis MAIN prog	•			Plan
146 TEAPLOT Terrain Encounter Analysis PLOT	•			Plan
157 C-TLAPOST CAT customized TimeLine Analysis			•	Comp
180 IFACE90-AASPEM AASPEM_CAT to MDTOOL	•			Plan
181 IFACE90 C-TAC to MDTOOL Interface	•			Plan

(3) 29 - SWAT. A suggestion was made by the CDT to replace the VMS/VAX SWAT with a PC version of SWAT. Status: Awaiting the final assessment of SWAT at the end of the Test and Evaluation of Field Demonstration No. 1.

(4) 60 through 72 - Data Bases. Originally delivered data base structures in the INGRES format. These data bases were delivered empty. Since data bases are needed to support the new CSDP, these original data bases will be converted to Informix if they meet the requirements. If they do not meet the requirements, new data base structures will be created in Informix. Status: Awaiting feedback from the industry review to determine the necessary data bases and structure.

(5) 108 - CAT-customized Tac Brawler - (C-TAC). The use of this CSCI is not required for trial application or field demonstration. An assessment is required to determine the necessity of porting this CSCI. Status: Awaiting an assessment at the end of Field Demonstration No. 1.

(6) 126 - Advanced Air-to-Air System Performance Evaluation Model - (AASPEM). The use of this CSCI is not required for trial application or field demonstration. An assessment is required to determine the necessity of porting this CSCI. Status: Awaiting an assessment at the end of Field Demonstration No. 1.

(7) 127 - Tac Brawler Main (TBMAIN). The use of this CSCI is not required for trial application or field demonstration. An assessment is required to determine the necessity of porting this CSCI. Status: Awaiting an assessment at the end of Field Demonstration No. 1.

(8) 128 - Tac Brawler Summary (SUMAIN). The use of this CSCI is not required for trial application or field demonstration. An assessment is required to determine the necessity of porting this CSCI. Status: Awaiting an assessment at the end of Field Demonstration No. 1.

(9) 129 - Tac Brawler Interface (IFACE). The use of this CSCI is not required for trial application or field demonstration. An assessment is required to determine the necessity of porting this CSCI. Status: Awaiting an assessment at the end of Field Demonstration No. 1.

(10) 145 - Terrain Encounter Analysis Main (TEAMAIN). The use of this CSCI is not required for trial application or field demonstration. An assessment is required to determine the necessity of porting this CSCI. Status: Awaiting an assessment at the end of Field Demonstration No. 1.

(11) 146 - Terrain Encounter Analysis Plot (TEAPLOT). The use of this CSCI is not required for trial application or field demonstration. An assessment is required to determine the necessity of porting this CSCI. Status: Awaiting an assessment at the end of Field Demonstration No. 1.

(12) 153 - Timeline Analysis (TLAPREP). This CSCI is a script file that allows the six task level workload analysis programs to be run from the original RUN-TOOLS main menu. The DTM will replace this main menu by launching the CDS tools through the tool pull-down menu. Status: This replacement/upgrade will be implemented after Field Demonstration No. 1.

(13) 180 - AASPEM interface AASPEM-CAT to MDTOOL (IFACE90). The use of this CSCI is not required for trial application or field demonstration. An assessment is required to determine the necessity of porting this CSCI. Status: Awaiting an assessment at the end of Field Demonstration No. 1.

(14) 181 - C-TAC interface C-TAC to MDTOOL (IFACE90). The use of this CSCI is not required for trial application or field demonstration. An assessment is required to determine the necessity of porting this CSCI. Status: Awaiting an assessment at the end of Field Demonstration No. 1.

b. **Progress.** To maintain the current functionality of the software components, command files were converted to UNIX scripts. A C-shell environment was used for this task because it added the flexibility of aliases and an expanded command set.

The lexicals in the VMS-based command procedures were replaced by UNIX commands, or hand-coded, to provide the same function. Many of the command files had set-up symbols and logicals. Some of these symbols and logicals were converted under UNIX by setting environment variables. The most difficult part of this effort was to trace each of these symbols and logicals to the source to determine where they needed to be generated and exported in the UNIX scripts.

The majority of the original CSCIs were written in FORTRAN and supported by command files. Most FORTRAN code, which makes calls to system services and libraries, was modified to eliminate these calls. The SGI F77 compiler was used with a switch to accept certain commands, such as SMG$ and LIB$, to simplify the porting process. User-defined logicals created in the command files, such as file names in the FORTRAN code, were duplicated using the environment variable or soft link capability of the UNIX environment. The FORTRAN code, which comprised the majority of the tools, was very poorly structured. However, during this conversion, code re-structuring was held to a minimum because it did not affect the functionality of the program.

The order of the bytes in memory and on disk for the UNIX host is completely reversed from the VAX host. In the VAX, the least significant byte of the value is stored at the lowest address memory of the item. In the UNIX, the most significant byte of the value is stored at the lowest address memory of the item. This makes direct transfer of binary files impossible. Also, floating point values are stored differently on the VAX and UNIX machines. By using records, representation clauses and some bit-level operations, the conversion between floating-point types was accomplished.

The packing of data is different on the two systems. The VAX uses a Complex Instruction Set Computer (CISC) architecture and the SGI machines use a Reduced Instruction Set Computer (RISC) architecture. To attain the performance in a RISC processor, the architecture usually requires data items to be aligned on specific address boundaries, for example, a 16-bit value at an address divisible by 2, a 32-bit value at an address divisible by 4, and a 64-bit value at an address divisible by 8. The CISC architecture usually does not have this restriction.

The major challenge in converting VMS to UNIX was that the libraries used for file I/O on VMS are FORTRAN-based, whereas on UNIX they are C-based. Also, on UNIX, FORTRAN is preprocessed to the C-programming language. This caused a delay in the implementation when certain assumptions were made, specifically if uninitialized variables were assumed to be zero and local variables were assumed to be static. The successful approach was to ensure that certain variables were initialized to zero by using common blocks or the SAVE statement for local variables that needed to retain their values. Further limitations in the supported types of file I/O on UNIX required changes to the OPEN and FORMAT statements in order to handle I/O properly.

After porting from VMS to UNIX, the tools were tested and verified to make certain that they performed identically under VMS. To accomplish this, an identical input file on both systems was used for each tool ported. Porting was successful if the files generated by the tool were the same on both the UNIX and the VMS. To test the operation of the tool, the output from the SGI platform was used as input to the next tool in succession. The files generated by the tools under UNIX were compared with those generated by the tools under VMS to verify proper operation. If the tools tested did not generate valid output for a report under VMS, the problem was investigated under VMS and when the output was correct, testing was continued under UNIX.

5.3 Tool Development

During the reporting period, the analysis, review, and development of the computer-aided tools that make up the CDS continued. Equipment upgrades were made to improve computing capability, numerous changes were made to bring performance in-line with current industry standards, and evaluations were performed to establish new requirements. Eleven software tools were examined, as noted in the following subsections.

5.3.1 Engineering Design Simulator

a. Background. The EDSIM includes a number of major elements that support the concept of a rapidly reconfigurable breadboard cockpit simulator. The major elements of the EDSIM are the cockpit simulator base; support structure; seat system; cockpit controls and displays; front panels; interface electronics and a removable canopy; a McFadden Hydraulic Control System; a manager station with audio; a communication system with a video monitor and recording system; a test manager's console workstation that supports aircraft and airspace software simulation programs; a double-wide equipment rack that houses a programmable analog and digital input/output signal system (located next to the simulator); a power video and signal distribution chassis (located next to the simulator); nine display processors; and two types of Local Area Networks (LANs).

This section discusses the tools developed for the EDSIM during the current reporting period. These tools were required so that the EDSIM could be used for rapidly prototyping cockpit designs. Through the use of a scalable hardware and software architecture, the tools enhance the speed and efficiency with which the simulator can be reconfigured. The EDSIM was originally delivered in two distinct hard-coded cockpit configurations. When a hard-coded cockpit design is modified, the entire body of the software must be examined and each item that contains aircraft-specific data and functions must be replicated and modified for any change to the design. This time-intensive effort often results in the creation of new errors. The scalable architecture, implemented during this reporting period, allows designers to analyze new cockpit designs quickly and easily, retrofit existing cockpits, and add workstations and new simulation technologies without reprogramming. This section discusses the two most important considerations in building a scalable architecture: (1) a rapid prototyping design, and (2) replicated shared memory (also known as reflective memory).

b. Progress. The progress made on the EDSIM rapid prototyping design software and the replicated shared memory are discussed in Paragraphs 5.3.1.1 and 5.3.1.2, respectively.

5.3.1.1 Rapid Prototyping Design

The software architecture to enable rapid prototyping of cockpit controls and displays was implemented to provide a layered approach to cockpit development. There are three layers in the architecture: (1) the Simulation System Software, (2) the Simulation Application Software, and (3) the Cockpit Application Software. Figure 5.3.1.1-1 shows the three software layers that provide a simplified standard interface for integrating applications for flight simulation.

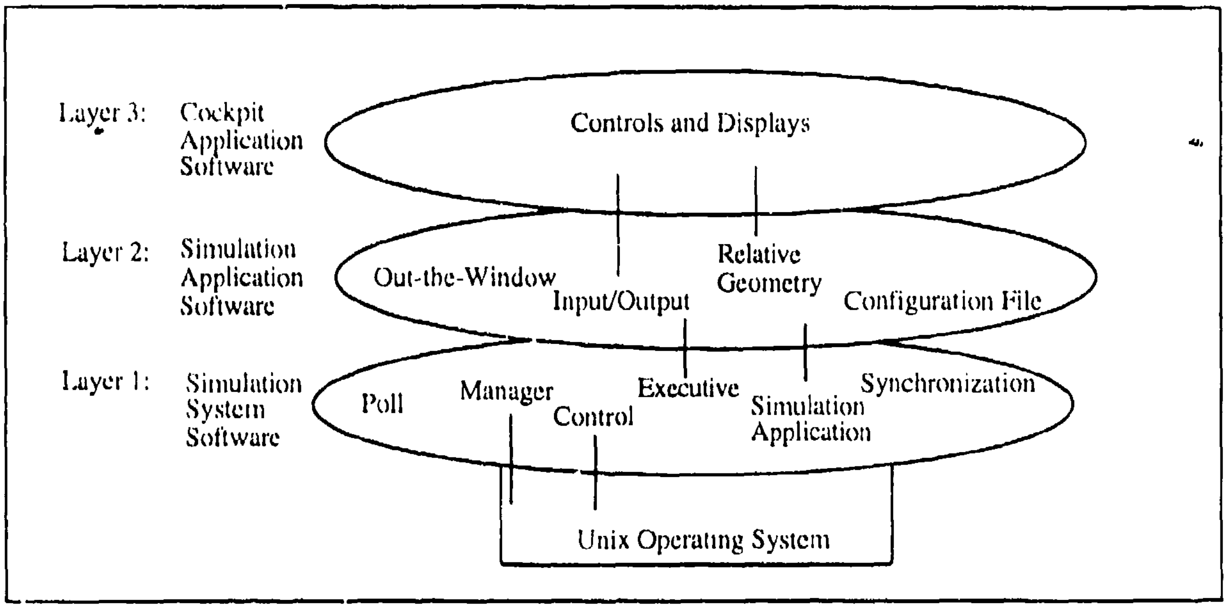


Figure 5.3.1.1-1. Three Software Layers

The simulation software discussed in this section is stored on one partition called ***bsim***. Every SGI has a *bsim* account that logs into the *bsim* partition. The workstation on which a simulation program executes is determined by the configuration file sent to the *manager* program. This configuration file specifies the host name to execute on, account to login to, program to execute (including path), any command line options, and finally a host name that designates where to send an output.

a. Simulation System Software. The simulation system software is the lowest layer in the software architecture. Programs at this layer control the overall simulation environment that manages software objects, controls simulation start and stop functions, and routes messages to the upper layer applications by way of message queues. The characteristics are:

(1) A simplified user interface to the simulation. At the lowest layer, each program calls routines to identify itself to the simulation, to identify the data objects to use or generate, and to check the state of the simulation. This program references data objects as existing in shared memory that can be in either true shared memory or replicated shared memory.

(2) Automatic routing of data between programs based on replicated shared memory technology and TCP/IP. This characteristic allows programs to be moved from one computer to another without reprogramming. It also provides performance improvements without recoding by adding larger or additional computers to the simulator, or by upgrading a computer to use replicated shared memory.

(3) Global Synchronization. This characteristic provides a more accurate clock for the tagging of data for the flight data recorder and allows the correlation of simulation data with externally collected data, such as a human physiological parameter collection system.

(4) Standard command line parameters to the applications. These parameters specify frame rate, source of simulation, and debugging options; generate standard statistics; and designate which cockpit the program is supporting.

(5) Standard output and standard error text. This text is automatically routed over the network to one workstation to provide a centralized display of simulation state information.

(6) A centralized mechanism for starting all simulation components from one workstation. The host, account, program name and parameters are specified in a configuration file so that a change to the configuration file will run a simulation component on another computer.

(7) A centralized control of the simulation state. This characteristic controls the simulation state from one workstation while, at the same time, allowing a distributed simulation.

Programs that reside in the Simulation System Software layer are the Manager *(Mngr)*, Poll *(Poll)*, Control *(Ctrl)*, Executive *(Exec)*, and Simulation Application *(Simapp)*. Each of these program names is preceded by *bsim* for this design, e.g., *bsimmngr*.

Mngr is the visual user interface for the simulation. The major functions of *Mngr* are to initialize, start, stop, pause, resume, and terminate simulations. *Mngr* uses a TCP/IP socket interface to send simulation state control messages to *Ctrl* and to receive current simulation state messages from *Ctrl*. *Mngr* reflects the characteristics described in items 5, 6, and 7 above.

Poll is a forked process that is begun by *Mngr*. The basic functions are to read and parse the configuration file, remotely execute the programs specified in the file, poll the standard output and standard program errors through the provided socket, and direct the collected output to *Mngr's* standard output. Standard output is linked to a console window on one workstation. *Poll* reflects the characteristics described in items 5 and 6 above.

Ctrl employs a TCP/IP connection to receive simulation state control messages from *Mngr* and to pass simulation state messages to *Mngr*. A similar TCP/IP connection links each *Exec* in the simulation and receives data object request messages and passes simulation state control messages and data object location messages to and from *Exec*. *Ctrl* manages the location of every data object defined in the simulation and allocates such objects to replicated shared memory, or commands *Exec* to provide a TCP/IP connection for the moving of the object's data from one computer to another. *Ctrl* reflects the characteristics described in item 2 above.

Exec is a program that runs on each of the simulation host computers. It has a TCP/IP connection to *Ctrl* to receive simulation state control messages, pass data object request messages from the applications, and receive and process object data location messages to pass back to the applications. *Exec* interfaces with each application on the same computer by way of a message queue. It passes messages to and from the applications and, in most cases, provides information on the status of the message it receives from *Ctrl*. *Exec* manages the locally instantiated data objects for the applications and informs the applications where shared memory is located so that the applications may bind to the appropriate memory segment. It also generates simulation sync to applications that request it, receives simulation sync by way of replicated shared memory, and sets up TCP/IP processes to move simulation data from one computer to another if replicated shared memory is not available on either computer that has, or needs, the data. *Exec* reflects the characteristics described in items 1, 2, and 3 above.

The *Simapp* library is linked to each simulation application. The library connects the application to the simulation and assists in the location and generation of data objects. It delivers simulation state control messages to the user-written portion of the application, delivers an indication of simulation sync to the application, and moves simulation data in and out of the application. It also parses standard command line parameters for frame rate, configuration, debugging and statistics information, and identifies the source of simulation sync. The library provides a message queue that passes data object request messages to *Exec*, receives simulation state control messages, and receives data object locate messages. The library reflects the characteristics described in items 1, 2, 3, and 4 above.

b. Simulation Application Software. The Simulation Application Software is the second layer in the software architecture. These applications are for configurable components that provide the same basic functions regardless of the aircraft/cockpit being simulated (e.g., the throttle, stick, and attitude indicator). The second layer components control the interface to the device and standard computing models. The characteristics are:

(1) Configuration File. The simulation components are selectable through a user-specified configuration file. The simulation configuration file is unique for each second-layer application. This allows prototyping by changing a file instead of changing the source code.

(2) Hierarchical Constraints. The simulation components make use of the lowest layer in the software architecture, and therefore are largely free from system dependencies. The only exceptions are the simulation components that control hardware. These components must reside on the computer that is interfaced to the hardware device.

(3) Small Well-Defined Components. A large number of small, well defined components allows the simulation to take advantage of the symmetrical multiprocessing capabilities of the SGI processors. While additional processors do not improve the execution speed of large monolithic programs, they significantly accelerate the execution spee. of a larger number of smaller programs.

The second-layer components are as follows:

(1) Out-the-window software. This component is common to all simulations. It was isolated as a second-layer application, both for modularity and for ease of replacement in the future.

(2) Digital and analog I/O to the cockpit. This component interfaces with the device driver. It distributes and collects analog and digital I/O to and from shared memory.

(3) Configuration File. The configuration file specifies how to distribute and collect the analog and digital I/O data to and from shared memory, and how to scale the data before it is deposited.

c. Cockpit Application Software. The Cockpit Application Software is the third layer in the software architecture. The applications are cockpit-specific; that is, they provide the functions for aircraft and cockpit design that cannot be supported generically by the configurable components of the second layer. Control and display logic is also an example of third-layer components. The characteristics are: (1) Configuration File, (2) Hierarchical Constraints, and (3) Small, Well-defined Components.

These characteristics function the same as similar ones in the Simulation Application Software described above, except that the components make use of data already available in the second layer instead of the lowest layer.

5.3.1.2 Replicated Shared Memory

The replicated shared memory network configuration was installed in the CDS. A two-node SCRAMNet demonstration system was initially installed and a series of tests were run before the decision was made to purchase the SCRAMNet. SCRAMNet has been in production and on the market for three years, and complete interface software and source code was provided with the system.

The system was easily integrated into the EDSIM architecture and successfully provides the distributed memory and communication function as expected. Previously, this communication overhead had to be handled by the v1sg10 processor. This no-overhead system increased the CATBATS execution time by twelve percent and decreased dropped frames by ten percent when compared to the execution performance documented in the Assessment Report (Reference 20).

Two other shared memory systems were examined: SmartNet and VMIC. Neither system was considered acceptable for installation in the CDS. The VMIC shared memory system has been on the market for less than a year and has not been proven in use. The SmartNet system is in development and is not available for purchase. Neither VMIC nor SmartNet have a software interface that would be compatible with the SGI system. If either system were used in the CDS, the software interface would have to be coded in-house or contracted to the system developer. Detailed information on all three systems was provided to the government and discussed in technical interchange meetings.

Tests have demonstrated the performance benefits of distributing large software programs across several machines to improve real-time performance. Installation of the SCRAMNet and testing required 40 labor hours.

5.3.2 Design Traceability Manager

a. Background. The original DEN did not perform any of its intended functions because it was never developed. As a result, off-the-shelf products were surveyed to determine if they could satisfy the necessary requirements as documented in the DEN documentation (Reference 21). A product called the DMCS was given consideration as a replacement for DEN, but it was discovered that although DMCS could be used to support the configuration management of design drawings, it could not fulfill all of the original requirements of the DEN (e.g., it could not differentiate between users and projects, keep track of multiple users working multiple projects with multiple tools, etc.). Since no satisfactory off-the-shelf products were discovered, a new CSCI was defined and named the DTM. The rationale and requirements for the DTM are detailed in the DTM Design Document (Reference 17).

The DTM was designed to assist the CDT in applying the CSDP during implementation of crew system design projects, and in tracing the progress and rationale behind the design decision. The need for the DTM was clearly demonstrated during the cockpit design activities performed in Delivery Order 10 of the previous CCCD Program (Reference 14). The list below contains the requirements that were partially completed during this reporting period. The DTM currently:

(1) Is the primary means to access integrated CDS capabilities.

(2) Is mouse-driven and functionally intuitive.

(3) Guides the CDT in the use of the CSDP by displaying it in the workspace.

(4) Supports documentation of daily activities through the electronic logbook feature.

(5) Provides a means to store design requirements throughout the CSDP by implementing six of the Design Requirements Document (DRD) (Reference 22) features.

(6) Differentiates between users and projects.

(7) Keeps track of multiple users working multiple projects with multiple tools.

(8) Provides project management support for the crew system design project.

(9) Provides a means to easily upgrade the CSDP.

b. Progress. Paragraphs 5.3.2.1 through 5.3.2.8 detail the design modifications to the DTM and the current implementation status of the DTM. One of the achievements in the DTM development was the establishment of the various types of traceability; these types include project and context traceability, intermediate and final product traceability, design and decision rationale traceability, and user history traceability.

5.3.2.1 Initial Design Traceability Manager User Interface

The initial DTM layout involved writing an X-Windows application in the C-programming language incorporating the standard Open System Foundation (OSF)/Motif and X-Toolkit Intrinsics libraries. This method of creating X-Windows applications is complex and time-consuming. To make simple visual changes in the X-Windows application, several lines of code have to be modified. Once the code is changed, the program has to be compiled, linked, and executed to test whether or not the changes are correct. The difficulties encountered with this approach, complete with the deficiencies in Wingz (Paragraph 5.3.2.2), led to the consideration of the alternative user interface approach described in Paragraph 5.3.2.3.

5.3.2.2 Wingz

Wingz is a commercial software product from Informix that is an "easy-to-use, high performance graphical spreadsheet that includes Hyperscript." Wingz was purchased to establish a programming environment in which graphical user interfaces could be developed and integrated with the Informix DBMS using the Hyperscript language and the Data Link utilities within Wingz. However, each time a Wingz-produced tutorial program attempted to connect to the Informix DBMS, the Wingz Hyperscript routines inadvertently disconnected the Informix DBMS server engine program, and caused it to stop running. The server engine program must run constantly to service requests from the DBMS user applications. After several weeks of trial and error, the Wingz technical support group claimed that the SGI version of the operating system was at fault. After receiving this information, the decision was made to review other software packages that would speed up development time and would not require special operating system upgrades or patches.

5.3.2.3 UIM/X and Informix 4GL Initial Integration

The need to accelerate development time for X-Windows applications created the need to review the various GUI builders available on the market. Before the decision was made to use the GUI-builder UIM/X software product, Bluestone Consulting technical support helped to verify that UIM/X could generate X-Windows applications that could be integrated with the Informix DBMS and would not create unexpected results. With this information, the UIM/X product was purchased (Change Proposal 5).

Initially, an example problem was accomplished that created an X-Windows application involving a Logbook Entry Form with UIM/X. The next step involved establishing a test Logbook

Data Base with Informix utilities. The 4GL code was written to accept parameters from the stack to insert a row into the Logbook Data Base. Functions were also written in the C-language and 4GL that used the UIM/X library to insert data into the data base. Once the ability to pass information into the data base was working, the ability to retrieve and display information from the data base was tested and established. The UIM/X product generates code that does not interfere with the Informix DBMS server engine program. UIM/X also reduces the time needed to develop the graphical look and appeal of X-Windows applications.

5.3.2.4 Software Installation

After selection of the Informix relational DBMS products (Section 3.2.7), the Informix Standard Engine (SE) data base server software was installed on the v1sg16; however, it was later upgraded to Informix OnLine data base server. The advantages that Informix OnLine has over the Informix SE are: (1) high performance; (2) high availability; (3) data consistency with the use of fault-tolerant mechanisms; (4) distributed data base access; (5) large-scale support of data bases with the use of shared memory caching and communication; and (6) multi-media data management capabilities. When the Informix SE was installed, the Informix ESQL for the C-Programming Language, the Informix 4GL and the Informix SQL software products were installed on the v1sg16 machine to allow design and development of the DTM and other applications such as the TMT and the GIT.

Since the Informix OnLine data base server is only installed on one machine, any user trying to run an Informix application that queries the data base server must initially log into v1sg16 machine. An Informix connectivity software product, Informix-STAR (a product of Informix that connects several data bases across a network), used in conjunction with the Informix OnLine data base server, establishes distributed data base support. Distributed data base support allows manipulation of multiple data bases at different network locations as if they were one common data base. Therefore, Informix-STAR is needed to separate the Informix users and the data base server software on different machines. With the Informix-STAR software product, any user of the Informix tools and corresponding applications can access the INFORMIX OL data base server without having to remotely log into the v1sg16 machine. Informix-STAR is scheduled to be purchased next year.

5.3.2.5 Methodology Data Base Design

For DTM to fulfill the requirement of guiding a CDT in the use of the CSDP, a data base to store the process was deemed necessary (Methodology Data Base). After reviewing the CSDE to obtain a data base structure, and verifying that the CSDP would conform to the same structure as the CSDE, it was determined that only one data base would have to be created to store the information for the CSDP and the CSDE.

This data base structure contains three main relational tables that are entitled activities, procedures, and technicals. The activities table contains the process activities; the procedures table contains the step-by-step procedures to be followed for each activity; the technicals table contains the technical contents of the product assigned to each CSDE activity. The information in these tables indicate specific CSDE fields, such as the Major Systems Acquisition Process (MSAP) phases (e.g., Concept Definition, Demonstration/Validation, Full-Scale Development, and Production and Deployment) and specific CSDP categories, such as Program Planning, Up-Front Analysis, Analysis, Design, and Evaluation.

5.3.2.6 Methodology Data Base X-Windows Applications

Section 5.3.2.5 explains how the DTM can guide the user through the CSDP. An associated requirement of the DTM is to provide a means to easily upgrade the CSDP. Once the design of the Methodology Data Base was established, the data base structure was initialized with Informix utilities. Next, the X-Windows applications were created to allow easy information entry into the data base.

Three main X-Windows applications were created to provide DTM with the ability to initially store and easily upgrade the CSDE and the CSDP. To enter the CSDE or the CSDP activities into the activities table, the Activity Form was created; to enter procedures into the procedures table, the Procedure Form was created; and to enter CSDE specific technical contents information into the technicals table, the Technical Form was created. In the future, these applications will be launched from within the DTM system administrator pull-down menu.

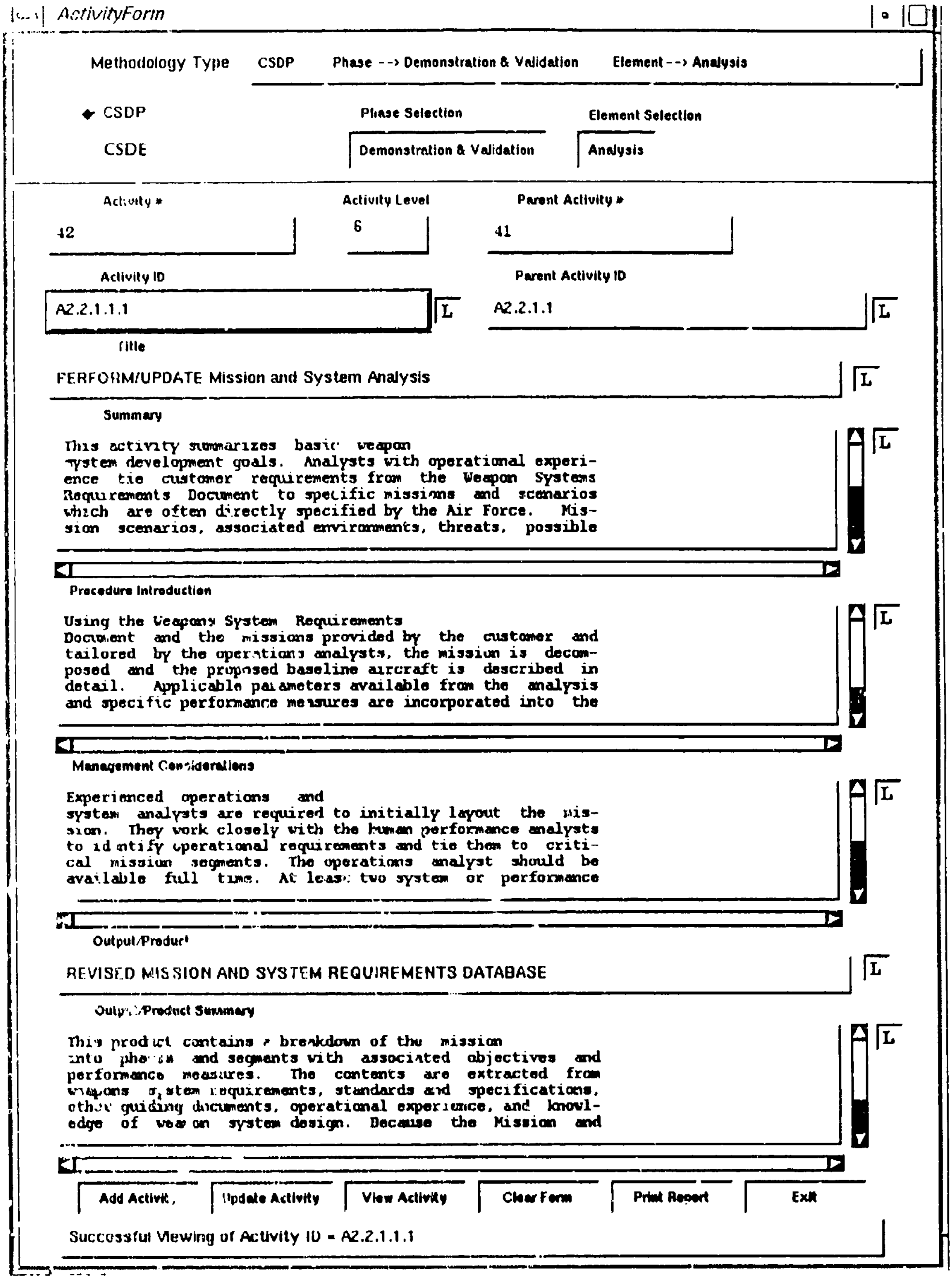
The Activity Form (Figure 5.3.2.6-1) is the most complex of the three X-Window applications. This application allows the DTM system administrator to add, update, or browse through the activities in the CSDE or the CSDP. In the screen capture of the Activity Form, this process is known as the CSDP. While using this Activity Form, the DTM system administrator specifies whether the CSDE or the CSDP is to be used by selecting the correct toggle button on the Activity Form. The following information is entered when adding or updating any activity in the Methodology Data Base: CSDE acquisition phase; the CSDP category; the identifier; the parent; the title; the summary; the CSDE management considerations; and the product fields. A 4GL report can be launched from this application by printing the current activities found in the Methodology Data Base activities table. New data from a separate word processor file that is running on the same SGI workstation can be cut and pasted into the Activity Form application to facilitate data entry.

The Procedure Form and the Technical Form (Figures 5.3.2.6-2 and 5.3.2.6-3) are similar in appearance and functionality to the Activity Form. The main difference is that the user must specify the activity to which the procedure or technical content is related, thereby establishing a relationship between the three Methodology Data Base tables. These applications are not as complex as the Activity Form application because less information is needed, due to the fact that the main information regarding each activity is already stored in the activities table.

5.3.2.7 Methodology Data Base Input

The CSDE Demonstration and Validation (Dem/Val) MSAP Phase was transferred from Macintosh disks to the SGI so that the Jot word processor could read the data. The information was cut and pasted from the Jot word processor into corresponding fields in the Activity Form, Procedure Form, and the Technical Form X-Windows applications and added to the Methodology Data Base. The CSDP was also transferred to the SGI and entered into these applications to update the Methodology Data Base in the same method.

The Methodology Data Base provides DTM with the capability to query and to display the CSDE or the CSDP information when the user is referencing the CSDE, or navigating through the CSDP within the DTM workspace. Having both of these processes, the CSDE and the CSDP, stored with the same data base structure allows the DTM to reference and update both processes in the same manner, using the same code. Since both the CSDE and the CSDP are stored within the same data base, and manipulated in the same manner, only one set of applications is needed to maintain and upgrade both processes. Additionally, the system administrator only learns the operation of one set of applications to be able to upgrade, review, or report information on both the CSDE and the CSDP.



ActivityForm

Methodology Type CSDP Phase --> Demonstration & Validation Element --> Analysis

◆ CSDP

CSDE

Phase Selection: Demonstration & Validation

Element Selection: Analysis

Activity #: 42

Activity Level: 6

Parent Activity #: 41

Activity ID: A2.2.1.1.1

Parent Activity ID: A2.2.1.1

Title: PERFORM/UPDATE Mission and System Analysis

Summary

This activity summarizes basic weapon system development goals. Analysts with operational experience tie customer requirements from the Weapon Systems Requirements Document to specific missions and scenarios which are often directly specified by the Air Force. Mission scenarios, associated environments, threats, possible

Procedure Introduction

Using the Weapons System Requirements Document and the missions provided by the customer and tailored by the operations analysts, the mission is decomposed and the proposed baseline aircraft is described in detail. Applicable parameters available from the analysis and specific performance measures are incorporated into the

Management Considerations

Experienced operations and system analysts are required to initially layout the mission. They work closely with the human performance analysts to identify operational requirements and tie them to critical mission segments. The operations analyst should be available full time. At least two system or performance

Output/Product

REVISED MISSION AND SYSTEM REQUIREMENTS DATABASE

Output/Product Summary

This product contains a breakdown of the mission into phases and segments with associated objectives and performance measures. The contents are extracted from weapons system requirements, standards and specifications, other guiding documents, operational experience, and knowledge of weapon system design. Because the Mission and

Add Activity | Update Activity | View Activity | Clear Form | Print Report | Exit

Successful Viewing of Activity ID = A2.2.1.1.1

Figure 5.3.2.6-1. Activity Form

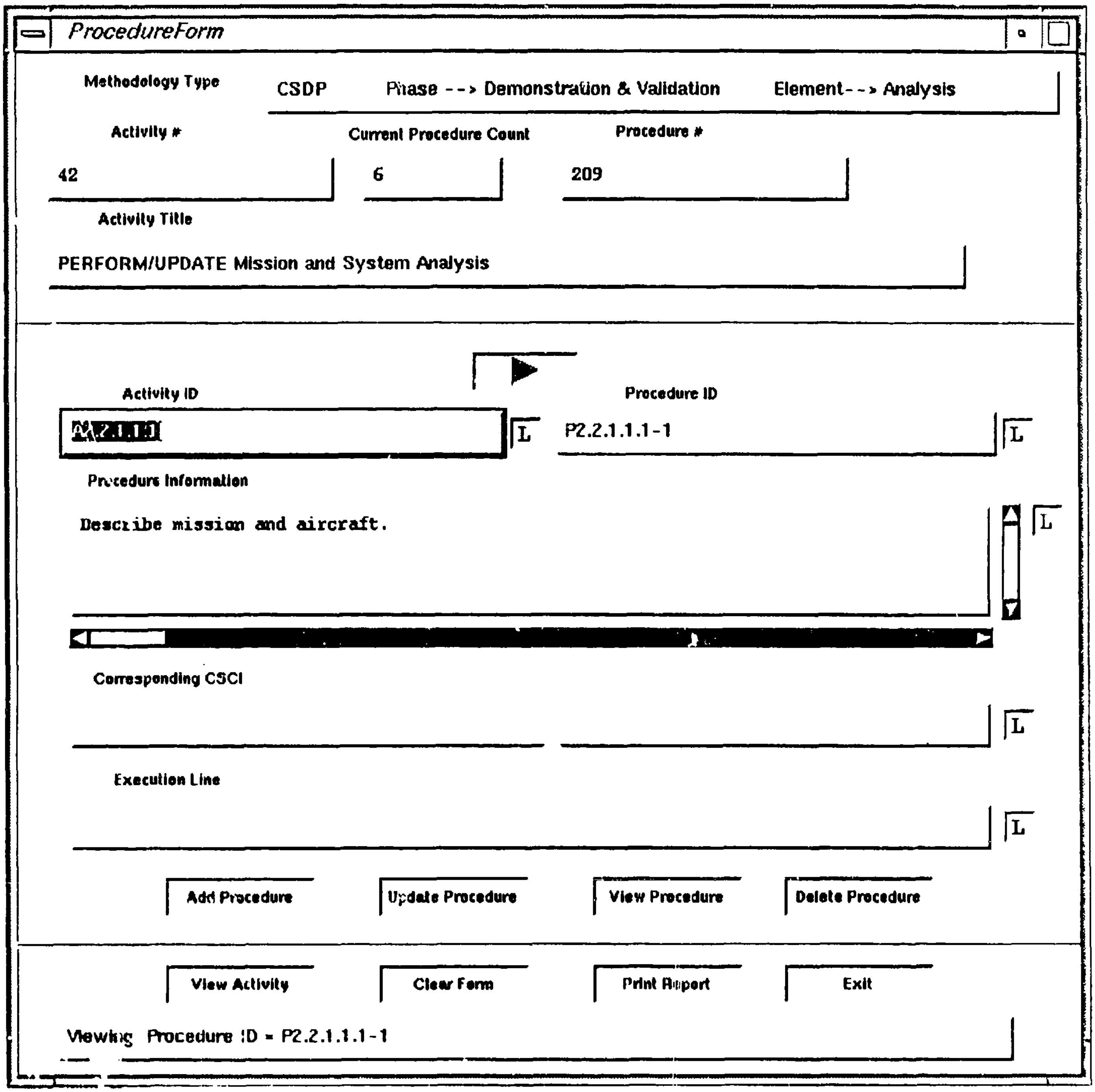


Figure 5.3.2.6-2. Procedure Form

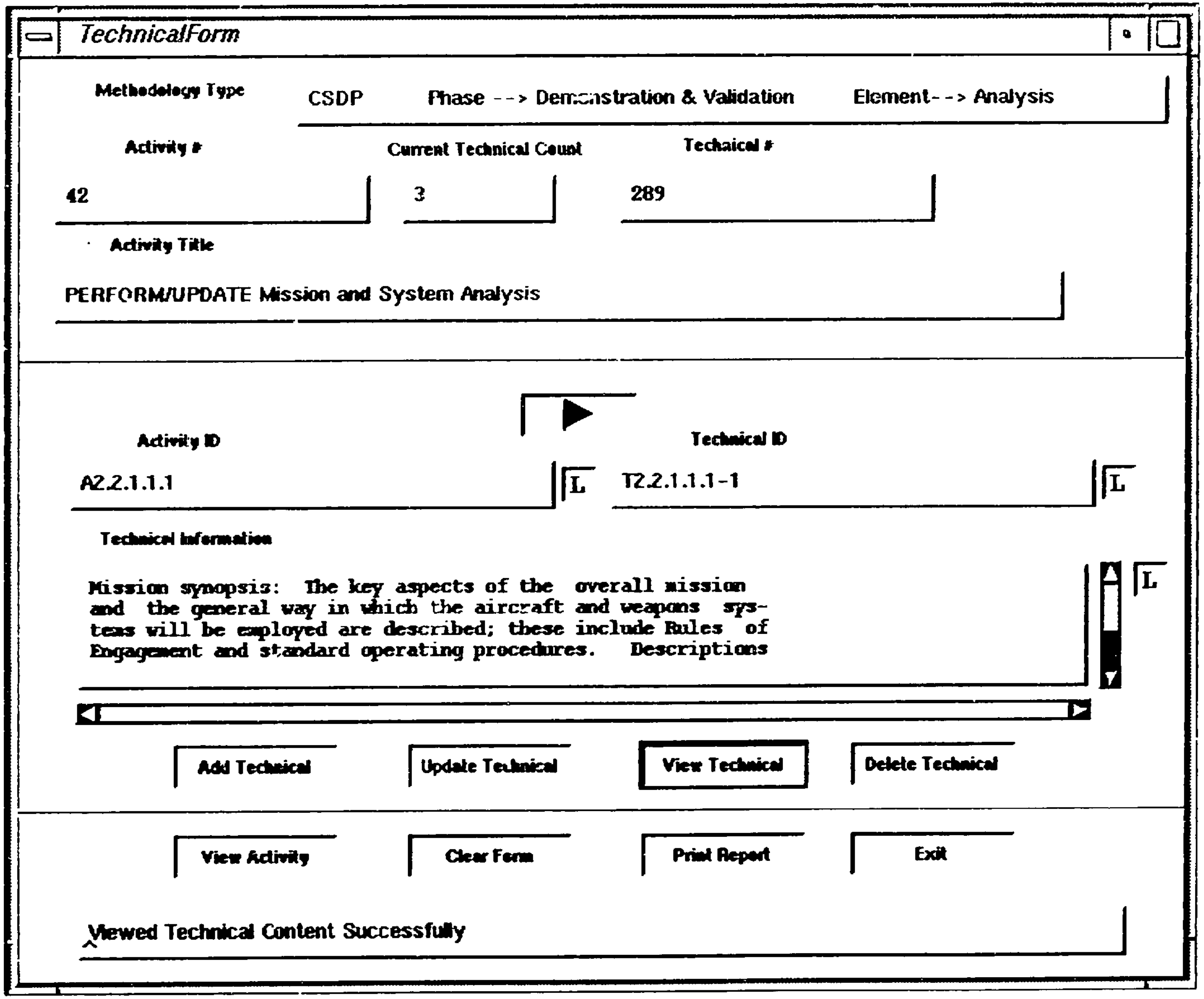


Figure 5.3.2.6-3. Technical Form

5.3.2.8 Revision of Design Traceability Manager Layout and Design

a. Additional Graphic Introductory Screens. An introductory graphical menu is now implemented in the DTM. This graphical menu provides options to view two information pages or to execute the DTM. One of the information pages provides an introduction to the purpose of the CDS; the other information page describes the DTM system requirements. The ability to launch DTM from the graphical introduction menu of DTM was implemented with Iris Showcase. The CDS objectives page and the format of the CDS system requirements page were also developed with the Showcase software and are not X-Windows applications. These pages are Showcase data files that need Showcase in order to be viewed in DTM. Iris Showcase was chosen for its ability to rapidly support full-color, bit-mapped graphical presentations, such as those featured in the DTM introduction.

Iris Showcase is a drawing tool for creating graphics with basic text-processing capabilities. Iris Showcase also has the ability to link other applications to these graphics. The difference between UIM/X and Iris Showcase is that UIM/X produces actual X-Windows applications, together with the executable code that can be integrated with Informix ESQL/C code to manipulate and query data base information. Also, UIM/X does not launch the applications that it generates. In contrast, Iris Showcase only creates graphical file images that cannot be integrated with Informix ESQL/C. The files that are created by Showcase are not stand-alone executable X-Windows applications; rather, they must be launched by the Showcase applications. The files that are created by Showcase do not provide the ability to accept data like the UIM/X generated applications do.

b. Main Menu Modification. The DTM user interface was enhanced by adding support for three distinctive types of users: (1) the DTM analyst/designer/evaluator, (2) the DTM project manager, and (3) the DTM system administrator. The DTM analyst/designer/evaluator has limited access within DTM because there is no update capability within the project management support utilities or any of the system administrative support functions . However, this type of user has the ability to browse project management data, launch CSDP/CSDE tools and navigate through the CSDP/CSDE, make Logbook entries, and complete product traceability forms. The DTM project manager has the same access as the DTM analyst/designer/evaluator plus access to an additional set of project management functions such as project creation, project team assignment, and task assignment. The DTM system administrator has more privileges than the DTM project manager, including the ability to modify or access the CSDE/CSDP and the users list; however, this access is not used daily.

The DTM main menu has a different set of selectable menu items dependent on the current type of user. Since the menu item accessibility of each user is identified, the DTM main menu was modified to incorporate these added functions. The DTM is currently implemented such that a single user can only update one specified project at a time. Another implemented feature is that a single user can have only one version of DTM running at any instance in time. The system administrator main menu selection items have been placed on-hold until the DTM project manager main menu and the DTM analyst/designer/evaluator main menu selection items are completely functional. Subsequently, the DTM project manager main menu and the DTM analyst/designer/evaluator main menu will be implemented. Currently, the X Window applications that are activated from these main menu selections are being implemented.

c. File Main Menu Selection. This is the most complex pull-down menu from the DTM main menu. Only the DTM project manager and the DTM system administrator will be able to use the full capability of this menu when development is complete. This menu will allow the DTM project manager to initiate a project, duplicate a project, add DTM designers to the current project team in each of the CSDP categories (i.e., Up-front Analysis, Program Planning, Analysis, Design, and Evaluation), and to assign and schedule activities to each DTM analyst/designer/ evaluator. At completion, the DTM analyst/designer/evaluator will have the ability to open an existing project and browse through the project schedule, the SOW, the DRD, the Project Plans, or the public directory of the current project.

The File pull-down menu is approximately sixty percent complete, while the menu-selectable items activate over forty separate X-Windows applications, the maximum number of applications is unlimited. The current count does not include applications that will be activated when the user wants to browse through the project plans because the project plan format still has to be defined. Currently, sixteen of these applications are completely functional. The layout of an additional twenty-four applications was initiated, but these layouts are not yet completely linked to the Informix DBMS and therefore do not manipulate data base information.

Three examples of applications launched from the DTM File pull-down menu of the DTM main menu that are complete are depicted in Figures 5.3.2.8-1, 5.3.2.8-2, and 5.3.2.8-3. The first figure illustrates the DTM application that is used to open a previously defined project and its corresponding context, a mission configuration combination. The second figure shows the DTM application that is used by project management to assign specific users to various categorized teams. The third figure details the DTM application that can be used to assign specific up-front analysis activities to each team member.

d. Activities Main Menu Selection. This pull-down menu allows the DTM analyst/designer/evaluator to view assignment on the currently opened project context. A project context is a specific cockpit geometry and mission flight configuration within the project. Project contexts are established when completing the DRD in the CSDP during an Up-Front Analysis activity. If the DTM analyst/designer/evaluator was assigned to more than one context on the project, a single context from the specified contexts available on the project must be selected. Once the activities list is displayed, an activity can be selected. If the user selects an activity from the selection box, the workspace of the DTM is updated. This method of updating the workspace makes it easier for the user to navigate through the CSDP or the CSDE to locate current project assignments.

The Activities pull-down menu is approximately seventy percent complete. Menu items actually activate distinct categories of the activities assigned to a DTM analyst/designer/evaluator. The ability to select from multiple contexts is not yet complete. However, when an activity is selected, the workspace is updated in regard to the current activity, information, and status section. The display of current procedures for selected activities is working for the following two specific activities in the CSDP: Perform Mission Profile Analysis and Complete the Design Requirements Document.

e. Process Main Menu Selection. If the user is the DTM system administrator, the Process pull down menu would allow the user to modify the data base tables with the Activity Form, the Procedure Form, and the Technical Form applications. The Process main menu option is only a selectable item open to the DTM system administrator.

The Process pull-down menu is approximately eighty percent complete. Applications that modify the Methodology Data Base are complete for the CSDE and some of the CSDP. Menu selectable items, however, are only existent at the DTM analyst/designer/evaluator and project manager levels. This pull-down menu is being modified to include the DTM system administrator items that activate the Activity Form, the Procedure Form and the Technical Form applications that manipulate the CSDE and the CSDP data in the Methodology Data Base.

f. Report Main Menu Selection. This selection will incorporate the ability to create reports based on the DTM project information stored in the Informix DBMS. Details

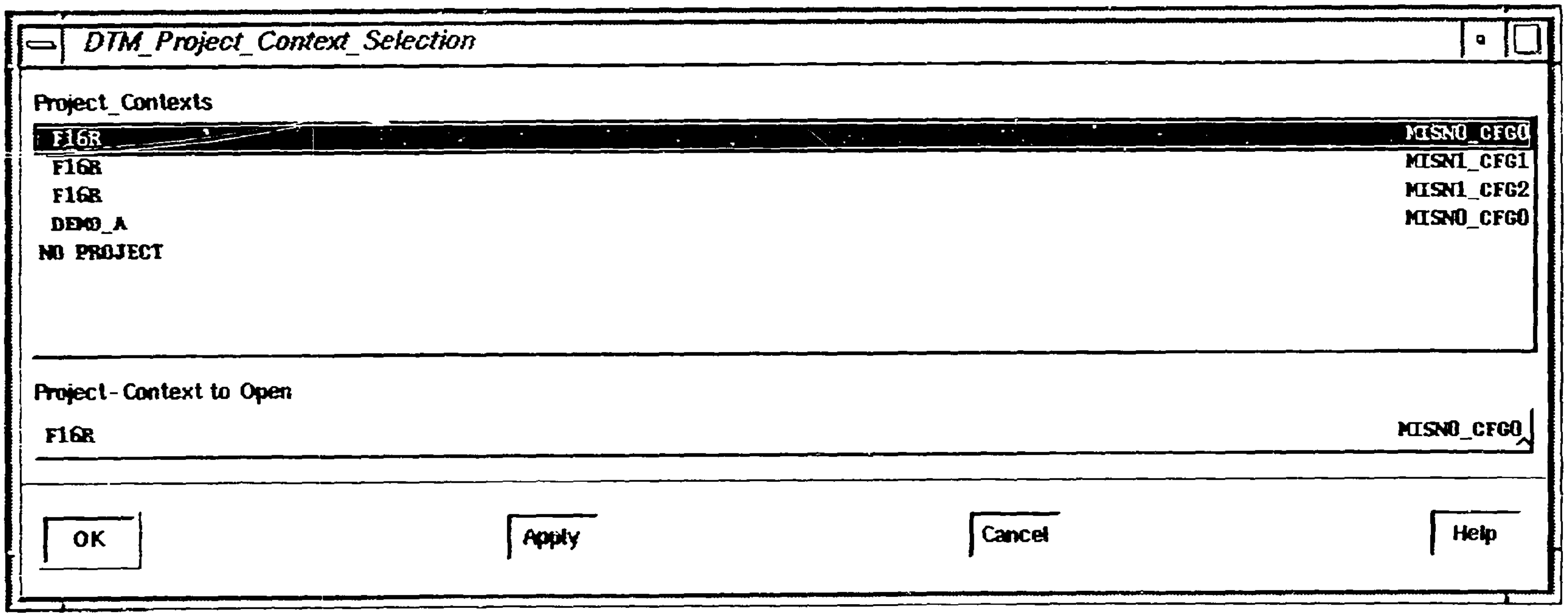


Figure 5.3.2.8-1. DTM Project Context Selection

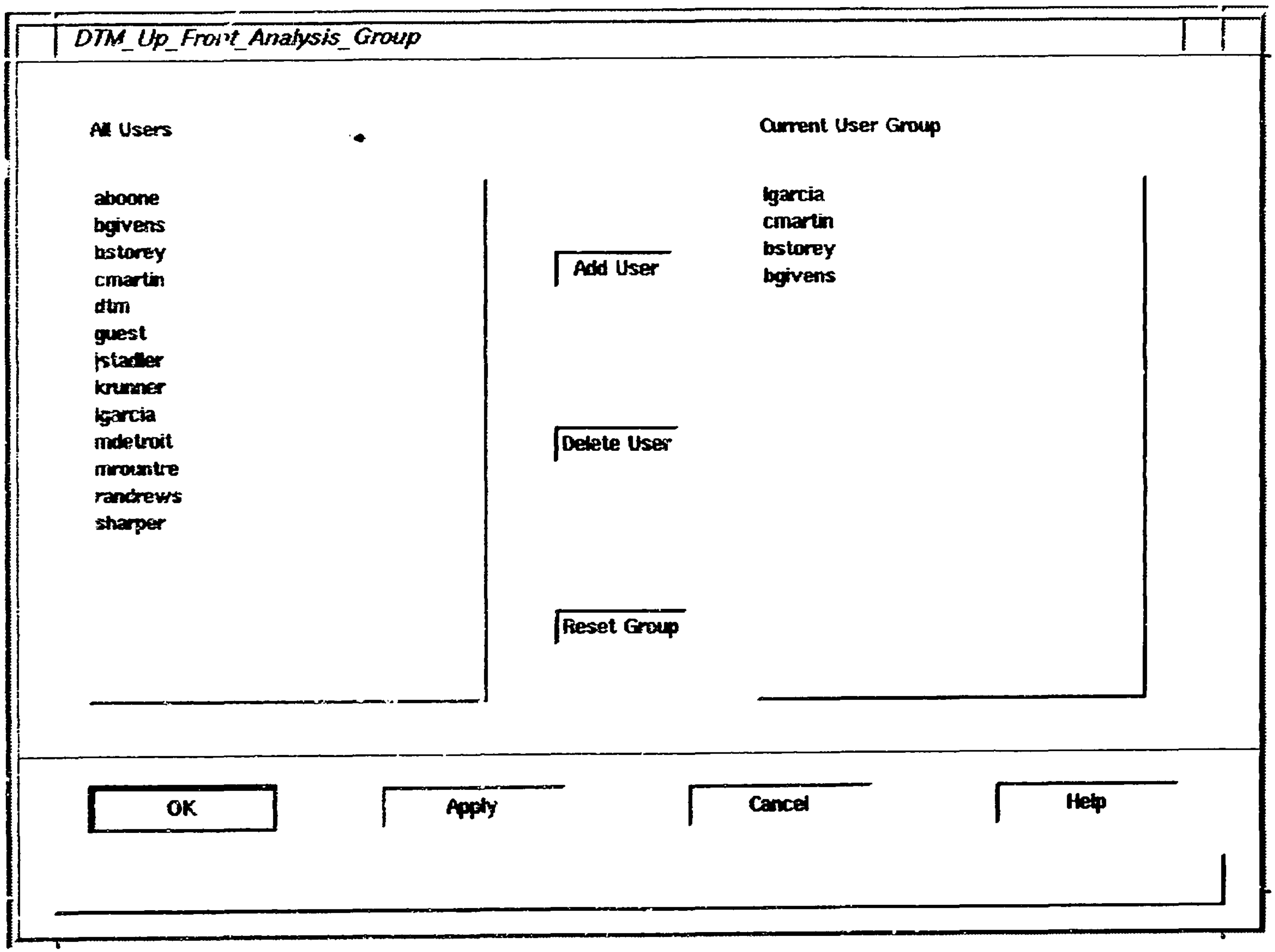


Figure 5.3.2.8-2. DTM Up-Front Analysis Group - Users

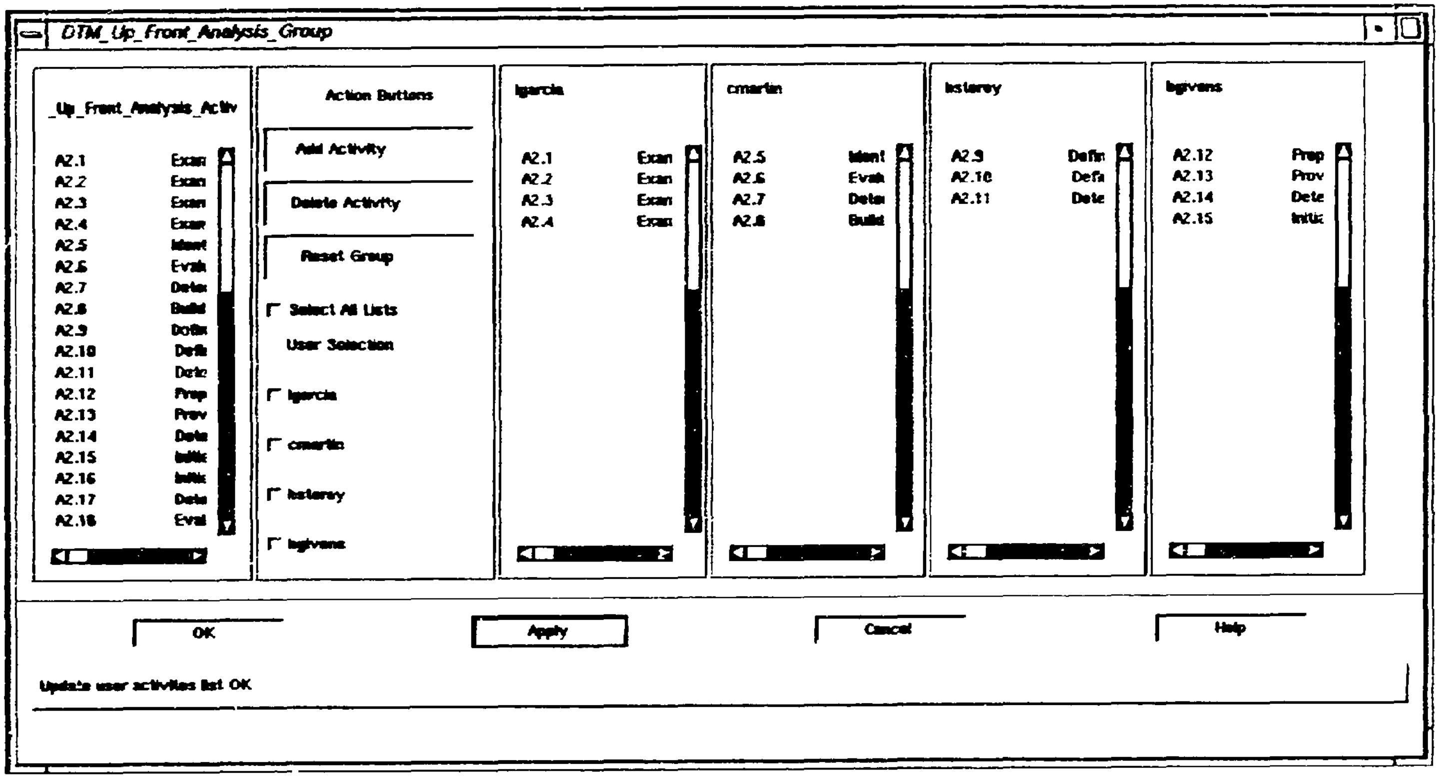


Figure 5.3.2.8-3. DTM Up-Front Analysis Group - Activities

about the report writer must be further defined. The need for this pull-down menu was identified; however, the design requirements must be defined for the report writer program that is launched from this menu item. The report menu has not been implemented, but its design is underway.

g. CDS Tools Main Menu Selection. This pull-down menu allows the user to launch the CDS tools from the DTM environment. The DTM system administrator can add, update, and delete the CDS tools from this menu when the applications are complete. When a tool is launched by the DTM, a set of background programs need to be launched to automate the tracking of file system modifications made by the launched CDS tool.

The DTM provides file-level creation/modification traceability to provide a method of incrementally documenting changes made to individual files throughout the CSDP. One of the background programs used to automate this method, known as the file tracer, was completed during this reporting period. The file tracer program was developed to scan one or more directories, identifying new or changed files. The default operation of the file tracer is to identify new or changed files with any user name. A list of user names can be given to the scanner to restrict the search. The file tracer has an option to recursively scan all sub-directories. The file tracer program stores the time each directory was last scanned to correctly identify new or modified files. Information about each new or modified file is stored in a data base and includes the name of the host computer (to support operation on multiple machines), file path, filename, file modification date and time, file user name, and file size.

After the file tracer program executes, it will launch a dialog box, the file prompter, that will ask the user to add the modified files to the public directory. Once these files are selected, copies will be made in the project public directory by another program so that other DTM analysts/designers/evaluators have access to these public files.

The CDS Tools pull-down menu is approximately fifty percent complete. The ability to launch several tools is complete; however, a spawner program needs to be developed to automate this procedure such that the data to launch specific tools is no longer hard-coded. The file tracer program is functionally complete and can track file modifications; the automation of this file tracer program is not yet complete. Currently it is run by selecting the user interactive button called *Intermediate Product/File Traceability*. The ability to obtain information from the output of this program and to prompt the user to copy files to the public directory is working, but it is not automatically called by the file tracer. Also, the file management calls to copy the selected files and products and to modify read/write/execute accessibility are not completely functional within DTM.

h. Reference Data Main Menu Selection. This pull-down menu allows the user to browse through any reference material that is stored in DTM. If the DTM system administrator is the current user, the ability to add, update, and delete this information is supported. Since the DTM system operator user functions are on hold, the ability to add, update, and delete reference material is not complete.

The Reference Data pull-down menu is ten percent complete. The ability to view the menu selection items exists and the ability to select the CSDE as a reference also exists; however, the workspace is only partially updated with information from the CSDE part of the Methodology Data Base when the selection of the CSDE occurs.

i. Help Main Menu Selection. This menu selection allows the user to view help information for DTM usage. If the DTM system administrator is the current user, he/she can add, update, and delete this information. The DTM system operator capabilities have been placed on hold; thus, the ability to modify the help information is not complete. Details about a help utility for DTM usage will be available once requirements are defined and the Help data base is populated. The Help menu is not implemented but its design is currently underway.

j. Current User Status Area. To provide a visual aid to the user, the current user status area displays the current project name, context name, user name, current process (i.e., the CSDE or the CSDP) being reviewed, and the CSDE-specific MSAP phase (e.g., Dem/Val) or the CSDP category (e.g., Up-Front Analysis).

The update of the Current User Status Area is approximately seventy-five percent complete. When the user runs the DTM, the user logon identifier is automatically updated. If the user has selected a default project context during a previous session, that context is automatically opened when the user runs the DTM. In this instance the status area is also updated with the project and the corresponding context name. The CSDE MSAP phase and CSDP category fields will be updated when the navigation of the CSDP or the CSDE in DTM workspace is implemented completely.

k. User Interactive Buttons. The addition of the intermediate/final product traceability and the User History Buttons allow additional interactive forms to be launched from the DTM. However, the traceability button may not be needed in the future when file product tracing is completely automated. The current status of the User Interactive Buttons is explained in the following paragraphs.

The Logbook Entry Form (Logbook, Figure 5.3.2.8-4)) allows the user to maintain a daily log of activities. The default mode is called *New*, and it allows the user to add another entry in the data base tables. Users can also search for specific entries, recall them for view or edit as authorized, and modify their own entries using *Search*. Also, reports can be generated using the same query mechanism used in *Search*.

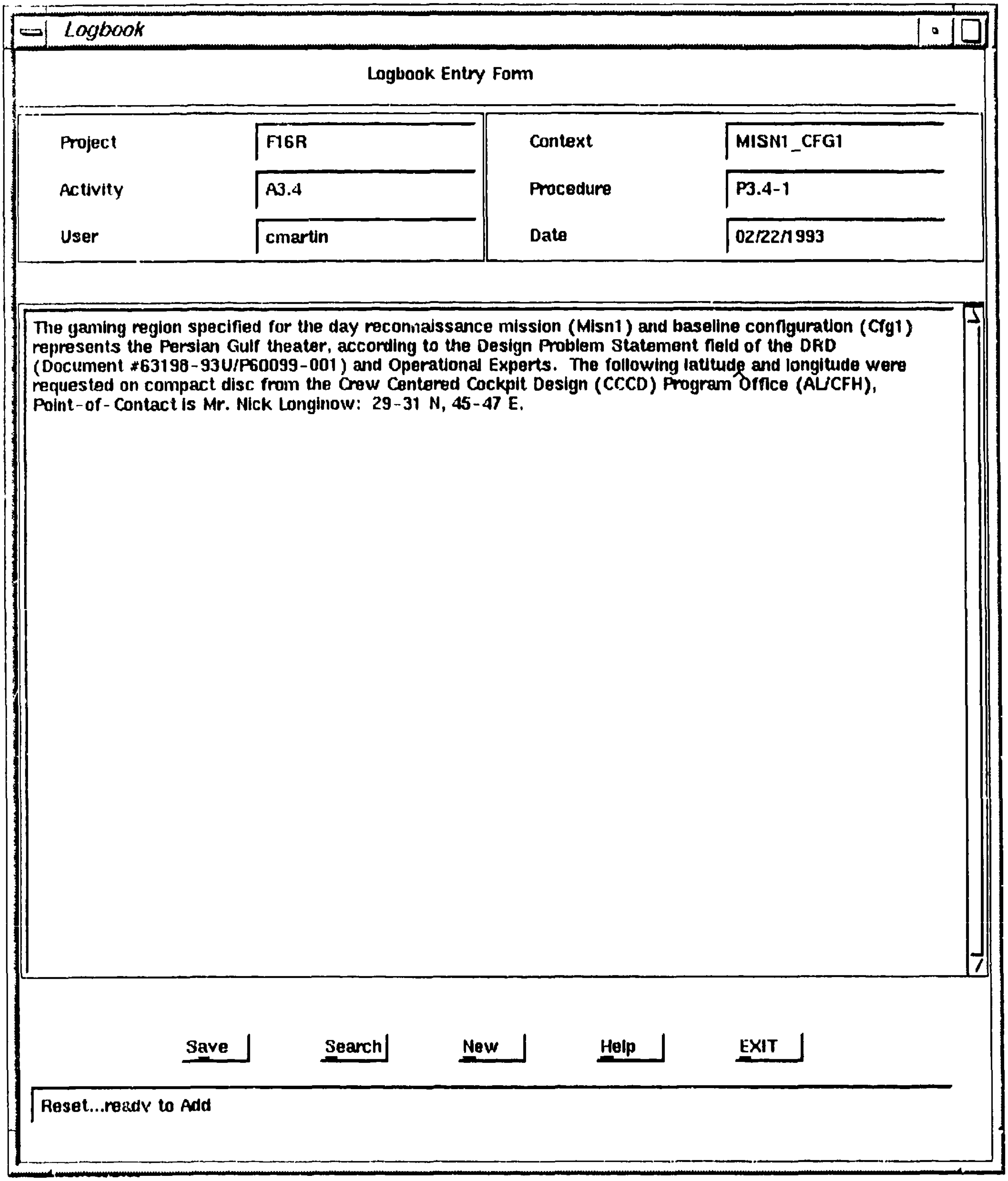
The Logbook is attached to its data base and all features except report generation are functional. *Search* currently only works for the user name and activity fields. The Logbook is approximately ninety percent complete. Extensive error checking must be added to this interface.

The Product Traceability Form allows the user to manage traceable file/products or deliverables. This form was built in this reporting period, but complete functionality is not yet attached. The data base structure is set up, but it is not linked to the form. When selected, the Product Traceability Form opens in *New* mode, and is ready for an entry to be input into the form. *Search* allows the view or edit of other data base entries as authorized by user name and user type. Report generation and extensive error checking will also be supported in the future. The Product Traceability Form (Figure 5.3.2.8-5) is approximately thirty percent complete.

The Lessons Learned Form allows the user to document useful information for future reference or inclusion in the general lessons learned data base. This form (Figure 5.3.2.8-6) is based on Air Force Form 1251. The Lessons Learned form was built, but complete functionality is not yet attached and therefore it does not query or manipulate any data base information. The Lessons Learned form is approximately thirty percent complete.

The User History Button will eventually display the content of the user's history table from the traceability data base. Population of the user history table in the traceability data base is implemented when the DTM user completes significant events within DTM such as creating a new project or assigning activities to project team members. The User History Button is approximately forty percent complete. The application to display current user history information is not yet complete although some events implemented in the DTM do update the user history table in the traceability data base.

l. Navigation Buttons and Workspace. The DTM Main User Interface, Figure 5.3.2.8-7, shows the navigation buttons and the DTM workspace. The highest-level activity



Logbook

Logbook Entry Form

Project	F16R	Context	MISN1_CFG1
Activity	A3.4	Procedure	P3.4-1
User	cmartin	Date	02/22/1993

The gaming region specified for the day reconnaissance mission (Misn1) and baseline configuration (Cfg1) represents the Persian Gulf theater, according to the Design Problem Statement field of the DRD (Document #63198-93U/P60099-001) and Operational Experts. The following latitude and longitude were requested on compact disc from the Crew Centered Cockpit Design (CCCD) Program Office (AL/CFH), Point-of-Contact is Mr. Nick Longinow: 29-31 N, 45-47 E.

Save | Search | New | Help | EXIT

Reset...ready to Add

Figure 5.3.2.8-4. Logbook Entry Form

Product Traceability

Product Traceability Form

Project	F16R	Context	MISN1_CFG1
Activity	A3.4	Procedure	P3.4-13
User	mrountre	Modification Date	04/28/1993
Product Creation Date	04/06/1993	Baseline Date	04/26/1993
PTR Creation Date	04/28/1993	Approval Date	05/10/1993

Product Description

F-16R Event Timeline, filename 'PG.fdr' (for Persian Gulf flight data recorder file)

Product Location

Silicon Graphics - v1sg12
Directory Pathname - /magik_data/bats_user

Objectives

The objective for creating the F16 Recce mission profile event timeline (ETL) was to graphically depict the combination of activities representing the actual mission events and timing. Generating the mission profile event timeline is the first step in translating mission requirements into system (aircraft and crew member) requirements.

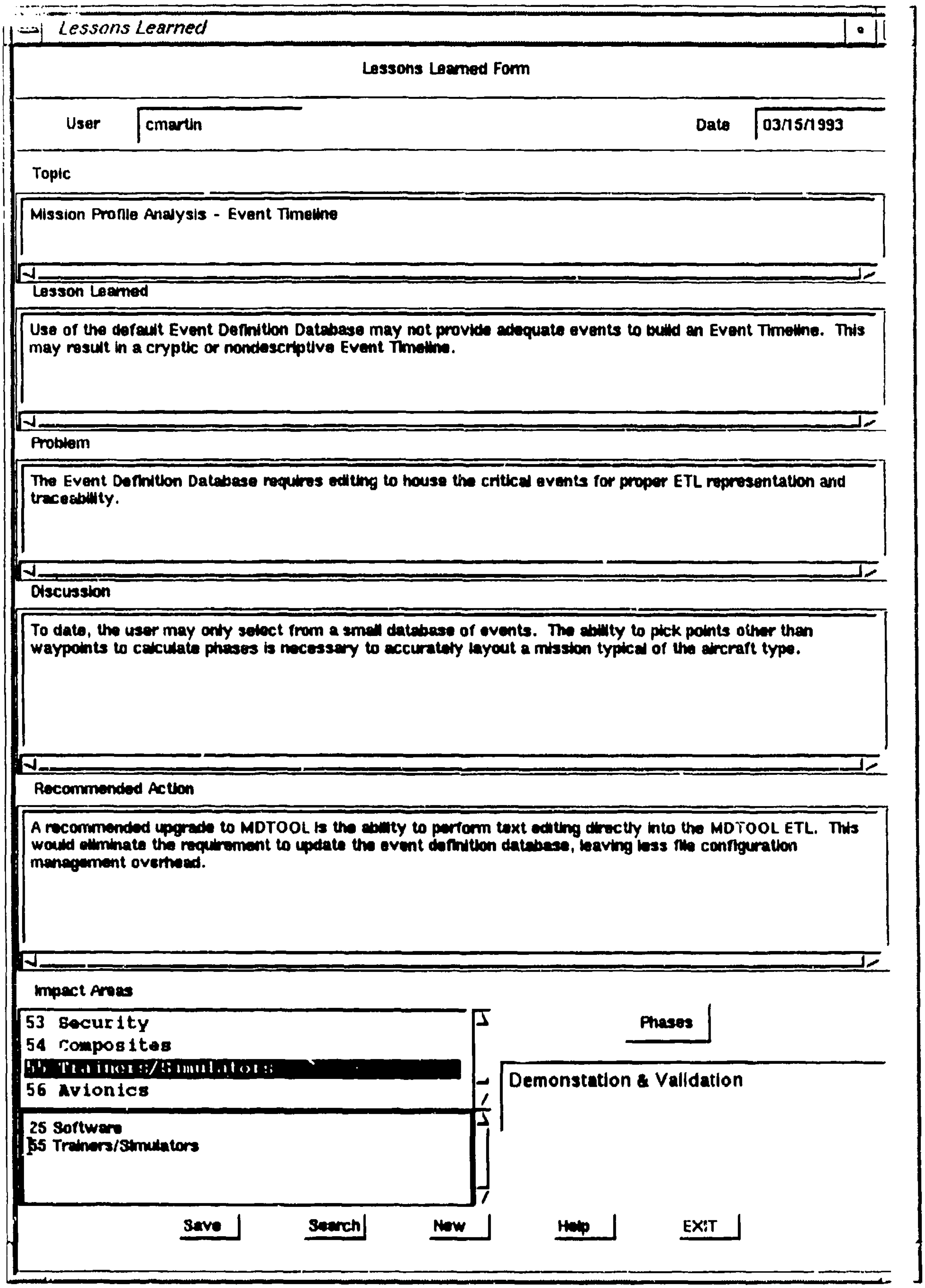
The Mission Parameter section of the DRD was used as the driving input for the generation of the F16R ETL.

Related Products

Design Requirements Document (Document #63190-93U/P60099-001)
Mission Scenario File
Simulation Test Plan

Save | Search | New | Help | Exit

Figure 5.3.2.8-5. Product Traceability Form



Lessons Learned

Lessons Learned Form

User: cmartin

Date: 03/15/1993

Topic

Mission Profile Analysis - Event Timeline

Lesson Learned

Use of the default Event Definition Database may not provide adequate events to build an Event Timeline. This may result in a cryptic or nondescriptive Event Timeline.

Problem

The Event Definition Database requires editing to house the critical events for proper ETL representation and traceability.

Discussion

To date, the user may only select from a small database of events. The ability to pick points other than waypoints to calculate phases is necessary to accurately layout a mission typical of the aircraft type.

Recommended Action

A recommended upgrade to MDTOOL is the ability to perform text editing directly into the MDTOOL ETL. This would eliminate the requirement to update the event definition database, leaving less file configuration management overhead.

Impact Areas

53 Security
54 Composites
55 Trainers/Simulators
56 Avionics

25 Software
55 Trainers/Simulators

Phases

Demonstation & Validation

Save | Search | New | Help | EXIT

Figure 5.3.2.8-6. Lessons Learned Form

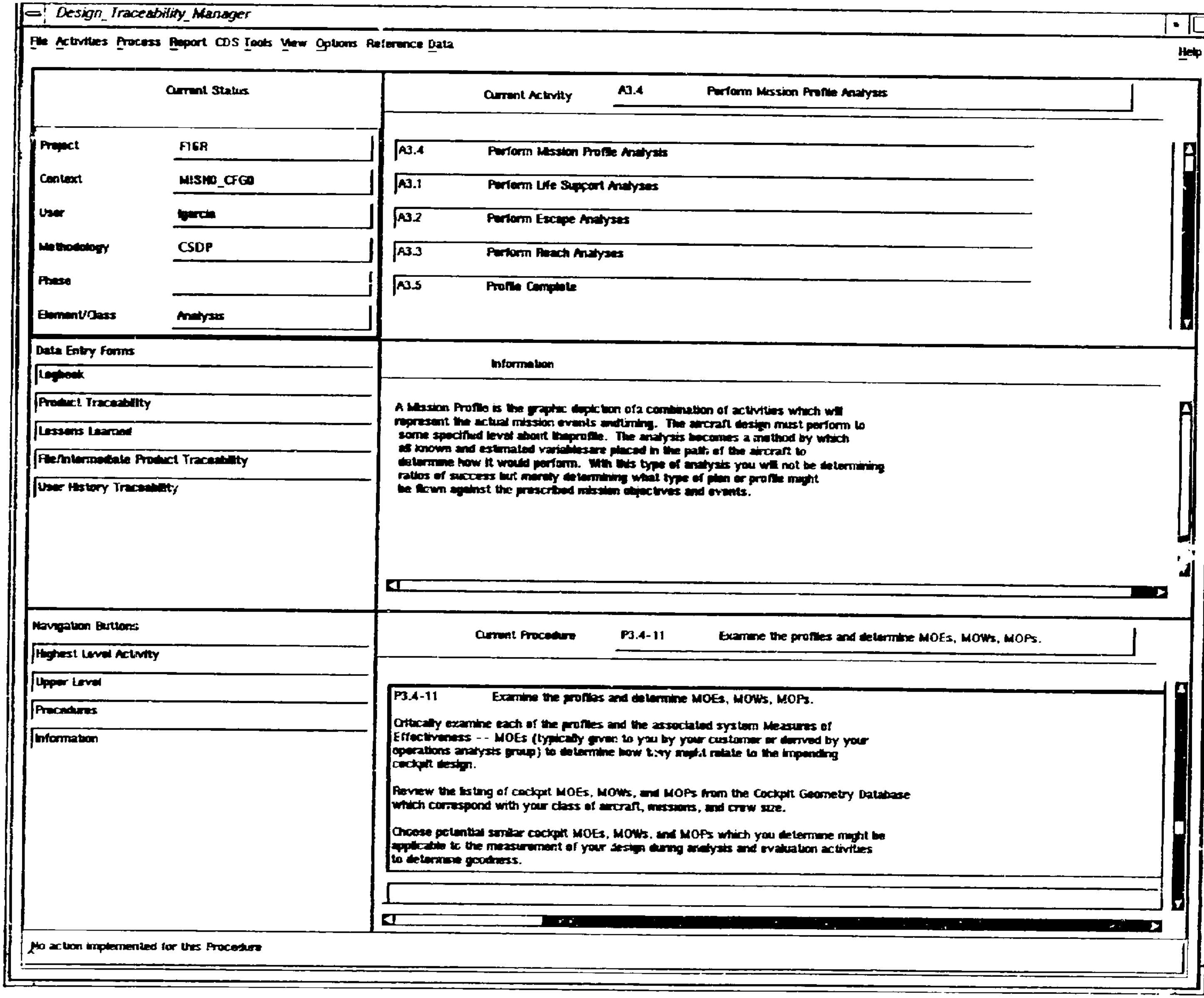


Figure 5.3.2.8-7. DTM Main User Interface

button allows the user to move to the root activity of the selected process, CSDE or CSDP. The user is automatically placed in the CSDP unless the CSDE is selected under the Reference Data menu. The upper- and lower-level navigation buttons allow the user to navigate easily through either process that is selected.

The DTM workspace is divided into three areas: (1) activities, (2) information, and (3) procedures. The activities area of the workspace displays the current activity and its associated activities. When an activity is selected, its title is put into the current activity area and the information area and the procedure area are updated with its information. The information area was added directly to the workspace to show the information page specific to the currently selected activity. The procedures are actually push-buttons with titles in the bottom section of the workspace. When a procedure is selected, it becomes the current active procedure. If the procedure involves another application or data entry, according to the CSDP, a user interface form would be launched upon the selection of a procedure button and the launching of its form.

The navigation buttons are approximately twenty percent complete. Once the workspace updates are completely functional from the activities pull-down menu, the navigation buttons will be functional in minimal time. Once the workspace is completely implemented, certain functions will be established to retrieve and update the workspace. These C-routines and functions will then be activated from various navigational buttons and will not need to be recreated.

m. Supporting Data Bases. The process, users, projects, status, log data, design requirements documents, and traceability data bases were created to support DTM and currently exist on the Informix DBMS data base server. These data bases will be updated as the implementation of the DTM continues.

5.3.3 Cockpit Automation Technology Battle Area Tactical Simulation

a. Background. Merit Technology Incorporated developed a unique configuration of their commercially licensed software program called the Battle Area Tactical Simulation (BATS). This unique configuration, now called CATBATS, includes a combination of custom-developed software and Merit's commercially licensed BATS software. (CAT stands for Cockpit Automation Technology, which was the original name of the CCCD Program.)

BATS is a simulation planning, execution, and analysis software package that comprises many components. It is a tool that is intended to be used to study aircraft combat engagements in simulated environments using digital terrain data (DTED and DFAD) from the DMA.

The host CATBATS program executes solely on an SGI Iris 4D multiprocessor workstation, such as the EDSIM Manager's Workstation (an Iris 4D/240 GTXB). CATBATS uses the Ethernet TCP/IP protocol to communicate between the simulation and graphics software processes. Merit Technology's MeritNET interprocess communication package allows all other CATBATS processes to communicate within the multiprocessor Iris environment. Selected run-time graphics programs execute on other Iris 4D single processor workstations. All of the necessary data files reside on the host Iris.

b. Progress. Several timing tests of CATBATS scenarios were conducted. Previously, CATBATS and the Simulation Control Logic Program (SIMCLP) ran on the same machine at just below 20 Hz for a one or two aircraft scenario. With the new softwa[illegible]ture, CATBATS is executed by a small shell program on v1sg10 and the input and outputs are transferred across SCRAMNet to v1sg17. This allows CATBATS to execute above 24 Hz for a one or two aircraft scenario while the display server program executes asynchronously at about 24 Hz.

At the beginning of the reporting period, CATBATS Version 5.31 was used, but it would not execute correctly under Version 4.0.4 of the SGI/UNIX operating system. Merit Technology recompiled the code under the new operating system and a new number (Version 5.32) was assigned. Merit Technology also responded to CR 203 to limit the process number that could be assigned to the aircraft models being executed during a simulation run. The maximum allowable number of processes that can be run is four, each of which can support five aircraft models. The user interface was modified to reflect this limit. This new software was assigned Version 5.33. At the end of the reporting period, Version 5.33 was still being used.

5.3.4 Timeline Management Tool

a. Background. The TMT (Timeline Management Tool) is used to elaborate time sequences of events to the elementary task level in support of mission decomposition activities. The TMT replaced the Network Management Tool (NMT) because the NMT suffers functional limitations that greatly impair its usefulness as a design tool. For example, to derive a model of aircrew member activities translated from mission requirements, the CDT requires support from Event-Function and Function-Procedure relationship data files for the decomposition of the Event Timeline (ETL). The NMT forces four hierarchical levels of ETL decomposition (Event to Function to Procedure to Task). This hierarchy of mission decomposition often fails to capture the most intuitive method of elaborating timeline elements. A more natural method is to allow any level of decomposition of events to obtain the task composition. It may be desirable to use more than four levels of elaboration to fully describe a given event to the task level. Large scale modification of the NMT was too complex because it was originally written in the List Processing (LISP) programming language and subsequently translated to Ada. A data type called an *access object* was defined to simulate the dynamic object creation and storage reclamation of LISP. The code that performs the emulation of the LISP storage management is heavily VAX/VMS dependent. Because elements of the *access object* are used throughout the NMT, porting it to the UNIX SGI host would have been too difficult.

b. Progress. The TMT Design Document (Reference 23) was created to outline TMT capabilities and user interface implementation details, and to provide information and examples about using the TMT program. The purpose of the TMT is to read a time tagged list of events from a file and allow a user to interactively elaborate it to any decomposition level, and write the results to a timeline data base. Its other main functions are to edit and write input files for originally developed analysis software (i.e., Mission Scenario Analysis [MSA], Information Analysis [IATOOL]).

The TMT is being implemented in three phases. The first phase is to create data bases using the Informix DBMS with the 4GL language to quickly implement the core functionality of the TMT. 4GL permits access to functions to provide simple text based menu and windowing capabilities, focusing on the task of quickly creating and debugging a data base oriented replacement for the NMT. The second phase is to add a user friendly X Window/Motif graphics user interface to the TMT. Completion of this step will bring the TMT user interface into compliance with other tools in the CDS software environment (e.g., DTM and GIT). The third phase is to verify that the TMT supports the original NMT relationships and interfaces.

(1) Phase 1. Use Informix DBMS capabilities (instaled on the SGI) for creating, searching, and modifying data files were used in the Phase 1 implementation of the TMT. The TMT can read an event timeline data file, elaborate it to four levels or a user defined number of levels (depending on the mode the TMT is running under), and store the elaborated timeline data base. In addition, the TMT can be used to edit a timeline data base, delete a timeline data base, or write an MSA input file (only when the data base is restricted to exactly four levels per [illegible]). The editor has powerful capabilities, including two methods of searching, moving, [illegible]

and deleting. The move, copy, paste, and delete functions operate on the current object, and all of its sub objects.

(a) Phase 1 applications were implemented using the Informix 4GL and its text-based user interface. Each of these programs that support timeline decomposition and editing was completed. The X-Window/Motif graphics user interface will be implemented in Phase 2. Each of the TMT functions work in 4GL with imbedded SQL commands for data base management, but the error checking has not been completed. Errors as minor as entering filenames that do not exist, or entering text when numeric input is expected cause the program to fail. Error handling will not be accomplished until the second phase of implementation because it is a function of the user interface software and would be a duplication of effort. The input file for the TMT is an Event Timeline in the American Standard Code for Information Interchange (ASCII) format that is the output from Mission Profile Analysis activities using the MDTOOL.

(b) The functionality of the TMT was implemented in five separate applications, the TMT, the Cockpit Configuration Editor (CCE), the Edit Event Timelines (EET), the Edit Event Definitions (EED), and the Information Analysis Tool (IATOOL) input file editor. Four of the applications are utility programs that give additional functionality to the TMT that was previously provided by NMT. The CCE reads, edits, and writes cockpit description files. The EET is used for editing event timeline files, the EED edits event definition files, and the IATOOL allows the user to create an IATOOL file.

(2) Phase 2. The second phase of the TMT implementation will consist of adding an X Windows/Motif GUI to the six programs comprising the TMT. The first step to convert each program will be to lay out the different user interface objects (labels, push buttons, menus, scrolling list boxes, and edit fields) on a screen, taking the functionality and user requirements into consideration. The 4GL programs will then be analyzed to determine what portions of the source code lines can be moved directly to the new X Windows to implement the user interface objects. Code to check for and handle errors will be added in parallel with the new X-Windows user interface.

(3) Phase 3. The third phase of the TMT development involves verification that the TMT supports the original NMT relationships and interfaces to other CDS CSCIs. TMT will be tested for accurately reading an ASCII ETL and c[illegible]ng an input file for the CDS MSA software program.

(a) The TMT will read an ASCII input ETL file or a Flight Data Recorder (FDR) file developed by MDTOOL. Verification will be made through display of the FDR file in MDTOOL and the display of that same file through the TMT workspace area. Events and time-tags will be matched.

(b) The TMT will produce an input file for the MSA program. Verification will be made by comparing the output from the SGI-based MSA program that used input data from the NMT and the TMT on two separate executions of the MSA.

5.3.5 Geometry Interface Tool

a. **Background**. The GIT (Geometry Interface Tool) program is a re-design of the originally delivered Cockpit Instrument Panel Layout Program (CIPLP). The CIPLP tool was assessed to determine what changes were imposed by the rehosting of CIPLP from the VAX/VMS platform to the SGI UNIX platform. As a result of the assessment, the decision was made to perform a re-design of the CIPLP. The justification and rationale for this decision were documented in the GIT Design Document (Reference 24).

The GIT tool supports cockpit analysis and design activities during the development of a cockpit. The GIT program will:

(1) Read Universal CAD files and display cockpit geometry specifications for zones, panels, and controls/displays

(2) Relate human factors performance data such as dwell time and reliability to each control or display type.

(3) Generate a geometry input file of controls/displays with the human factors performance data for input into crew system analysis software tools

(4) Update cockpit geometry and human factors data bases

The GIT functional flow is shown in Figure 5.3.5-1

b. Progress. The GIT tool is being developed using FORTRAN 77, ESQL/C, and the Informix DBMS. The GIT replaces the internal screen editor in the CIPLP program that is used for manipulating and saving cockpit geometry and human factors data in the VAX directory files. The Informix data bases are currently used to store this data, in addition to storing the developed ESQL routines that are used to access and manipulate data in the Informix data bases.

During the development of the GIT, the I-DEAS Version 6.0 CAD tool from SDRC is being used for 3D modeling of the cockpit. The geometry data from the cockpit model is written to a universal file produced by I-DEAS. The GIT tool extracts the necessary cockpit geometry data from this file.

The GIT permits the user to select an existing Informix data base to use or to create a new Informix data base file. The user is prompted for a data base name and upon entry of this name, the function opens an existing data base or creates and opens a new user data base.

The GIT main menu allows the user to select the other functions of the GIT such as reading and modifying the human factors data base, or merging the human factors data with geometry data. The user is supplied with a menu selection and is prompted for an input, after which the corresponding function is executed. The development of the X-Windows/Motif graphics user interface window entry forms is not yet complete.

The GIT extraction and transformation function is being developed to access the cockpit geometry data contained in a universal data file. The universal file is a standard interface that is used by all major CAD systems, including I-DEAS. Thus, the ability to read universal files frees the GIT from an exclusive dependence on any specific CAD system or on any specific version of such a system.

The I-DEAS universal file contains the 3D cockpit model that was created using the I-DEAS 6.0 graphics system. The purpose of this function is to search through the universal file and extract the geometry data of the cockpit. Once this information is obtained, it is necessary to perform rotations and translations of all initial cockpit points to obtain the final cockpit geometry data. This function pertains only to the I-DEAS universal files. The purpose of the GIT function that converts an IGES file into an I-DEAS universal file is to allow users of the CDS to obtain cockpit geometry data from any CAD system, e.g., CADAM or CATIA. The GIT informs the user how to convert the IGES file into an I-DEAS-created universal file. The necessary cockpit geometry data is then extracted from the universal file. There is a potential for universal file format

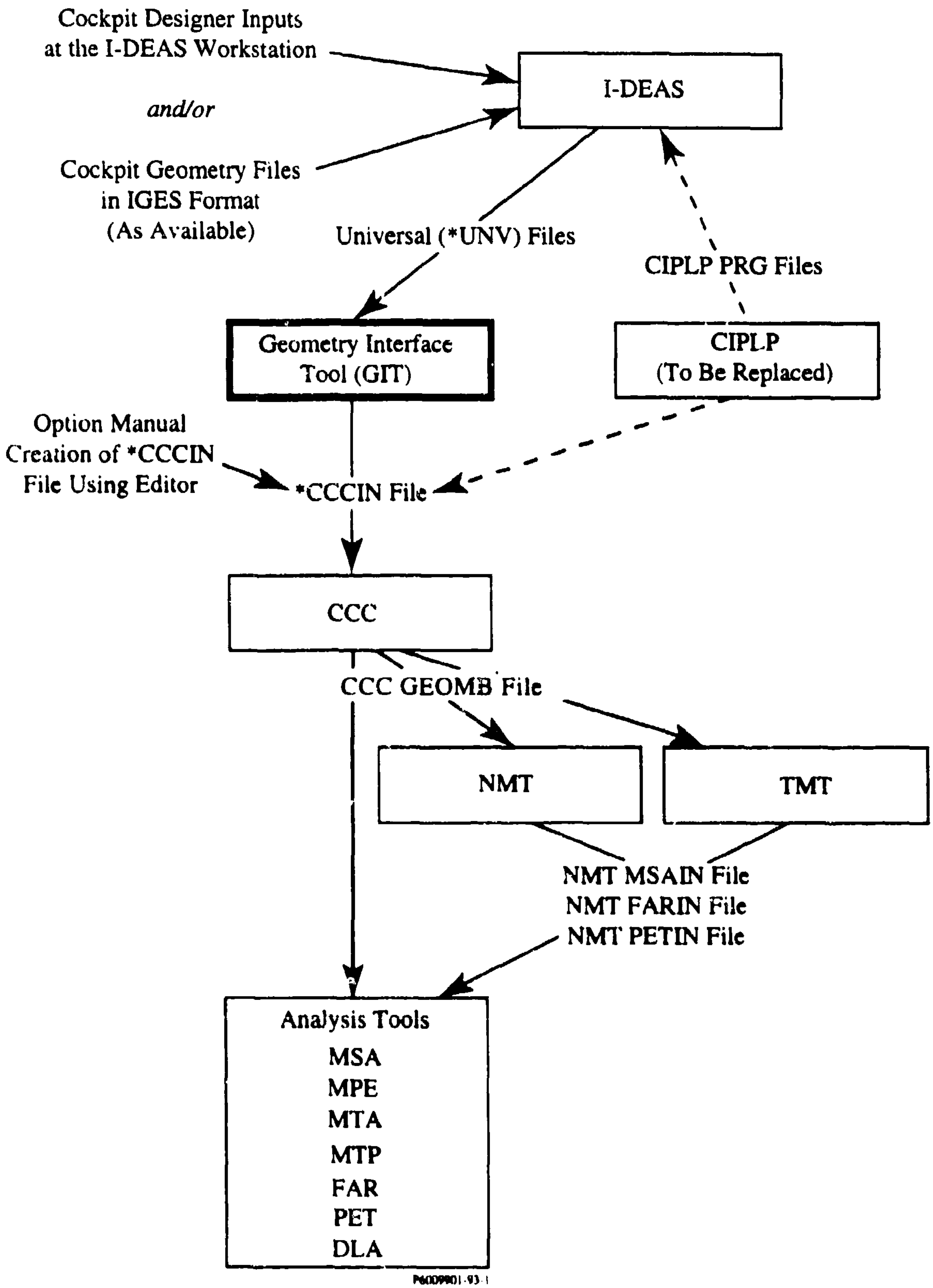


Figure 5.3.5-1. Flow of Geometry Data Using the Geometry Interface Tool

problems with later versions of I-DEAS. As a result, GIT will have to be upgraded to accommodate I-DEAS upgrades. The impact of performing a GIT upgrade during a project will be dependent on project time and budget constraints.

A separate Cockpit Configuration Control (CCC) Human Factors Data Base, which was part of the originally-delivered software, is maintained to contain information for all control, indicator, verbal, auditory, mental, and message items pertaining to the cockpit. The purpose of the *merge human factors data base function* is to incorporate selected items from the standard CCC Human Factors Data Base into the User Cockpit Geometry Data Base.

The *User Data Base input function* is being developed so that users can update and store data into the existing Informix data bases. This function allows users to enter operator data that is associated with the cockpit geometry data extracted from the CAD graphics file and the human factors data retrieved from the standard CCC Human Factors Data Base. The user is also able to update the cockpit geometry data and the human factors data.

The *generate CCC input file function* is developed to obtain cockpit geometry and human factors data from the User Data Base to create an output cockpit file that is used by the CCC analysis tool. The CCC analysis tool reformats and distributes this data to other originally-developed CDS analysis tools.

The *generate error message function* will display error messages pertaining to the execution of the GIT program.

The Window Entry Form design and functionality is defined and documented in the current version of the GIT Design Document (Reference 24). An implementation plan/schedule is being generated to identify planned GIT development and status.

When a baseline version of GIT has been developed, verification testing will begin. A baseline version is a version that meets the minimum capabilities of reading an I-DEAS universal file, reading and writing a human factors data base file, data basing cockpit geometry with merged human factors data, and writing an input file for the CCC program. The verification testing will involve: (1) writing a CIPLP input program file for I-DEAS and the CCC analysis tool, (2) having I-DEAS to write a universal file from this CIPLP program file; (3) having the GIT read the universal file, create a cockpit geometry data base, merge this data base with the standard CCC HF data, and write a CCC input file; and (4) executing CCC using both the CIPLP and the GIT files. The two CCC output files will be compared for matched output.

5.3.6 Mission Decomposition Tool

a. Background. The MDTOOL (Mission Decomposition Tool) is an interactive mission analysis, planning, and decomposition program used for accessing mission requirements, mission objectives, and performance measures and criteria. It enables the user to rapidly generate, store, retrieve, and modify data for air combat mission scenarios. Once the mission scenario is constructed and saved, it can be executed and viewed as the planned route is flown. An FDR file is generated on execution of the mission, which can be used in conjunction with a mission event timeline to analyze the mission. These timelines can be edited to insert pilot-generated events, and the entire event timeline can be used as an input file to the other CDS analysis tools.

b. Progress. During this reporting period, the MDTOOL was upgraded from Version 4.03 to Version 4.06. This new version, received in April 1993, had no file incompatibilities or software discrepancies that were common in previous software upgrades. It was fully tested to verify the following enhancements:

(1) A government-owned module of the MDTOOL (uiedit.c) generates an FDR file for use by the TMT. The FDR file is a time-based ASCII file of mission events.

(2) The color editor works properly.

(3) The user is prompted (cautioned) prior to saving an edited FDR file that a file with the same name exists. Saving the edited FDR file will overwrite new values to the existing FDR file if it has the same filename.

(4) Files can be saved that have blanks in the file name.

(5) A wind model does not have to be specified prior to the execution of the scenario.

(6) The FEBA is present during execution and playback modes.

(7) The gaming region is rescaled when it is redrawn during a timeline edit.

(8) Version 4.06 fonts are more legible in each of the interactive pull-down menus.

(9) Previously placed icons are visible during playback of the FDR files.

(10) The ability to add feature data (i.e., roads, airports, and buildings) to MDTOOL and BATS is available.

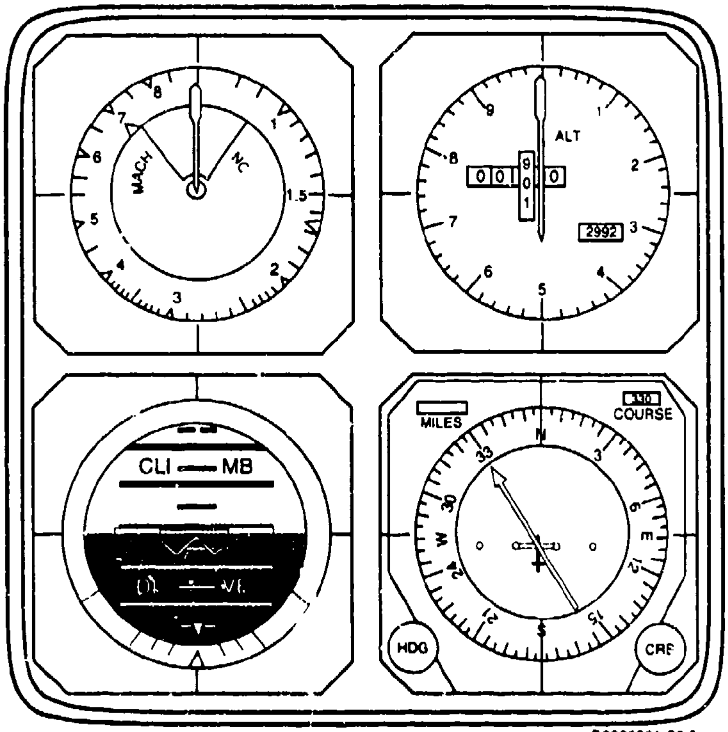
5.3.7 Graphics Modeling System

a. Background. The GMS (Graphics Modeling System) is the software tool that is being used to develop cockpit display formats and associated dynamics. GMS is also being evaluated to determine: (1) ease of use; (2) realism of visualization; (3) ease and speed of modifying displays; (4) ease of incorporating new and modified displays into the EDSIM; and (5) update rates.

GMS was easy to use, although the complexity of the display affects the amount of time required to develop the display. The appearance of the generated displays is acceptable. Care must be taken when modifying a display to ensure that the existing dynamics are not affected or deleted. It is possible to integrate the display and dynamics into the simulation in a short time frame. The main obstacle encountered in using GMS is the slow update rates of complex dynamics for time-critical displays. A possible solution for improved real-time performance is to run critical displays on higher performance workstations.

b. Progress. During this reporting period, GMS Version 4.0e was installed on the CDS and is currently in use. No problems were encountered in the conversion from Version 4.0d to 4.0e. The models developed in previous versions of GMS were usable without conversions.

The displays for Field Demonstration No. 1 are being developed using the SGI Graphics Libraries (GL), which enable better real-time performance than the X-libraries. The GMS-based programs for the Performance/Control Display (PCD) and five multifunction display pages were completed. The PCD is a suite of gauges consisting of the airspeed/mach indicator, altimeter, horizontal situation indicator, and the attitude direction indicator (Figure 5.3.7-1). After development of the PCD and integration into the simulation, the required 20-Hz update rate was



P6009901-93-5a

Figure 5.3.7-1. Performance/Control Display

not achieved. Table 5.3.7-1 indicates the type of performance that was seen with the display, using the specified number of gauges. The PCD code was sent to SL for help in optimization. The engineers at SL tried several optimizing techniques to improve performance. One such technique involved running the GMS-generated code through the GMS C-source code generator, thus eliminating GMS overhead. To use the code generator, all fill groups were removed from the model and a performance gain of 13.25 Hz was experienced. However, the model is unusable without the fill groups. Currently, GMS is being used as a rapid prototyping tool. When a candidate display is chosen to be flown in full field demonstrations, the display will be hard-coded to operate within the 20-Hz rate requirement. The multifunction display includes the master format page, Stores Management System (SMS) inventory page, reconnaissance (Recce) format page, Recce control page, and the manual depression angle entry page.

Table 5.3.7-1. Gauges Model Update Rates

NUMBER OF GAUGES PER MODEL	UPDATE RATE
1 (using X option)	6.4 Hz (Altimeter)
1 (using GL option)	10.0 Hz (Altimeter)
2 (using GL option)	7.5 Hz (Altimeter and ADI)
3 (using GL option)	4.3 Hz (Altimeter, ADI, AS/Mach Indicator)
4 (using X option and integrated into sim)	0.16 Hz (Complete PCD)
4 (using gl option and integrated into sim)	1.55 Hz (Complete PCD)

5.3.8 Sequitur's Workload Analysis System

a. Background. A trade study was performed that assessed the original workload modeling tools, along with several known and accepted workload models (Reference 25). The results of that study contained a recommendation to replace several programs that were used to calculate crew member workload with a single validated workload model called SWAS (Sequitur's Workload Analysis System), a commercial software program. The availability and applicability of SWAS was investigated and a copy was acquired for evaluation. The SWAS is a microcomputer-based application used for deriving operator workloads. The SWAS model is based on Wicken's Multiple Resource Theory (timesharing of concurrent tasks) for workload analysis. The workload calculation is derived through the simulation of an operator's task timeline in which the tasks have been assigned execution times based on Methods Time Measurement-derived values, channel of activity (i.e., visual, auditory, manual), and dependency on other tasks. This provides an estimate of workload by using a time-available/time-remaining paradigm, with provisions for timesharing capability (Reference 66, Appendix J).

b. Progress. The SWAS was used for workload analysis on the F-16R mission scenario. Currently, the only method to obtain graphical results of this analysis is to slave a Hewlett Packard (HP) laserjet printer directly to the PC hosting SWAS. A full report of the F-16R results is available and is included in this report as Appendix K (Reference 66).

5.3.9 Operator Assessment of Reach/External Vision Model/Computerized Biomechanical Man-model

a. Background. Both OAR (Operator Assessment of Reach) and E-Vision (External Vision, VAX-based software for assessing reach and vision) were replaced by COMBIMAN (Computerized Biomechanical Man-model). The OAR is a software program that calculates operator ability to reach panels, controls, or displays within the cockpit. The OAR defines four reach zones: Zone 1 - non-straining reach with shoulder harness; Zone 2 - operator straining against shoulder harness; Zone 3 - non-straining reach with waist harness; and Zone 4 - operator straining against waist harness at full stretching reach. The OAR output is a printed report of the data.

The E-Vision provides a means to generate vision envelope plots in the CDS environment using I-DEAS. The E-Vision plots monocular vision from the design eye point on Aitoff or rectilinear grids. The output plots are written out to IGES files for transfer to the I-DEAS drafting module to construct the vision plot overlaid on the grid.

The COMBIMAN was chosen as a replacement for OAR and E-Vision because it is a government-owned application that interfaces directly with I-DEAS, has increased capabilities, and is available for the SGI UNIX workstations. The COMBIMAN is an interactive, computer-graphics-based human factors evaluation instrument that supports analysis of: visual accessibility, strength for operating controls, reach capability with the arms and legs, and fit limitations/capabilities. The COMBIMAN gives the user selection from six combinations of Air Force clothing options and control over sizing the human-model, along with providing a number of alternatives for assigning and changing the dimensions of the model, including a set of multivariate models. The user may place the human-model into a drawing and analyze the interaction between the model's physical capabilities and the design elements related to the cockpit.

b. Progress. The COMBIMAN was installed and compiled in the C-CADS laboratory. The program linked and compiled successfully; however, the COMBIMAN cockpit geometry is not formatted for use in the COMBIMAN Crew Status Data Base. The version of COMBIMAN that was installed had a compatibility problem with the newer version of I-DEAS that is being run.

5.3.10 Quality Function Deployment (QFD)

a. Background. The QFD (Quality Function Deployment) is a system of related procedures and tools that enable a CDT to effectively interrelate customer needs with system requirements. The QFD Designer was proposed as a trade-off analysis tool. The QFD Designer capabilities were compared with the existing SUMMET model. One set of procedures and tools in the QFD area was identified as being superior to SUMMET, namely those within the AHP. The methodology of AHP aids in the ranking/prioritization of subjective attributes of a system.

The AHP provides a systematic approach to the tradeoff analysis needed for the CSDP. The AHP methodology may be used to prioritize design requirements or to evaluate design alternatives relative to specific attributes (for example, usability, reliability, and producibility). This process can be applied to types of trade-off studies involving subjective data. The process involves obtaining and analyzing paired-comparison data. The usability of the AHP methodology will be in trade-off analyses during Field Demonstration No. 1. Additional principles and procedures of QFD will also be evaluated further in this and future demonstrations.

b. Progress. The commercial software, QFD Designer, was procured from Qualisoft, Incorporated and hosted on an IBM PC-compatible workstation. This package is being considered for trade-off analysis techniques. A QFD training session was also received.

5.3.11 DI-3000 Graphics Software

a. Background. Several existing CDS analysis tools use the DI-3000 Graphics Software from Precision Visuals for the display of two-dimensional (2D) and three-dimensional (3D) workload plots. These tools previously required the VAX/VMS system.

b. Progress. Due to the conversion of the analysis tools to the SGI/UNIX system, an assessment was made to determine if the DI-3000 graphics routines could be replaced by the UNIX GL routines. It was determined that this replacement was not desirable because the routines were not compatible and considerable modifications were needed for the tools.

The DI-3000 software was acquired for use on the SGI/UNIX system. Installation on the SGI/UNIX system was difficult because the installation guide did not cover all of the steps necessary to install it on the SGI/UNIX system. Several DI-3000 UNIX script files and the source code were modified to make DI-3000 functional. Tools that were ported (the Mission Timeline Analysis (MTA), the Mission Task-time Probability (MTP), and the CAT-Timeline Analysis Tools (CTLA) that use DI-3000, are now functional on the SGI/UNIX system.

6. FIELD DEMONSTRATIONS

The CCCD Field Demonstration Program includes a series of five demonstrations in which the CSDP and the CDS tools will be applied and evaluated. These demonstrations will be performed through the use of a constantly improving set of activities, procedures, data bases, and tools that will increase the quality and consistency of the cockpit design process and products.

The F-16R Project was chosen as the first specific CCCD application. The development, application, and results of the efforts to date are discussed in this section. The designation F-16R is not an official Air Force term and is used herein to denote CCCD Field Demonstration No. 1. The remaining four field demonstration subjects have not been determined; therefore, no work was performed in this area during the period of this report.

6.1 Near-Term Fighter/Attack Systems: F-16 Manned Reconnaissance Mission

To illustrate a typical cockpit upgrade for an existing system, a modified F-16 that performs tactical reconnaissance was chosen as the design problem. The USAF is interested in a reconnaissance aircraft to replace the RF-4. The System Operational Requirements Document (SORD) (Reference 26) was published to document this need. The following requirements are excerpts from the SORD and will serve as guidelines in the performance of Field Demonstration No. 1.

a. The follow-on tactical reconnaissance aircraft will be an F-16 that is modified and equipped for the tactical reconnaissance mission. The approach is to modify existing F-16 C/D aircraft to a reconnaissance configuration. The F-16R will perform reconnaissance missions as either the weapon system's primary or secondary operational capability. The modifications to the F-16 will not delete or degrade the aircraft's ability to perform in the fighter/reconnaissance dual role capacity.

b. The F-16R must be capable of performing the full spectrum of reconnaissance missions, including day-night/under-the-weather imaging and medium and high altitude standoff imaging. The F-16R will be the primary platform employed by the tactical forces to provide tactical commanders with timely information of sufficient accuracy and detail to permit exploitation. The F-16R will be employed in fluid scenarios, beyond the first echelon, and under the weather, against mobile, fleeting targets where the human element increases mission success. In addition, the F-16R will be used to collect intelligence at times when the use of other systems is unsuitable.

The F-16R Project intends to take the above requirements and develop redesigned cockpit controls and displays to meet the required operational capability. This will include a combination of Up-Front Analysis, Program Planning, Crew System Analysis, Crew System Design, and Crew System Evaluation. See Appendix C (Reference 66) for an explanation of the major activities of the CSDP.

6.1.1 Up-Front Analysis

a. Background. The Up-Front Analysis category in the CSDP was created to give the CDT the ability to generate specific design requirements from top-level mission and system requirements. This area of the CSDP was defined but has yet to be implemented with complete procedures or tools. Potentially valuable tools, such as the QFD and Concept Mapping, were discovered through ongoing investigations to find the best methods to categorize requirements. The QFD methodology and tools were investigated to provide quantitative as well as qualitative design trade-off. The QFD methodology includes several tools and subordinate methodologies that are available for use, e.g., the QFD Designer, and the AHP. The QFD and AHP methodologies and tools are both used for trade-off analysis; Concept Mapping is used to assist in focusing information to generate appropriate mission and system requirements. Assessment of these methods will continue during the next several months.

At the beginning of the effort, it was apparent that procedures to perform all of the CSDP activities did not exist and could not be compiled due to time constraints. The decision was made to proceed with the Up-Front Analysis activities without predefined procedures, and to write the procedures for the Crew System Analysis activities that would be performed at a later date. In this way, a set of activities, procedures, and tools would be available to the CDT at the appropriate time.

b. Progress. The final output or product of Up-Front Analysis is a DRD (Reference 22) to guide the crew system analysis, design, and evaluation activities of the program. Paragraphs 6.1.1.1 through 6.1.1.8 discuss the Up-Front Analysis activities that were performed.

6.1.1.1 Design Requirements Document

An examination of the SORD and the General Dynamics F-16R ATARS Mechanization Document was performed by the operational experts to gain an understanding of the F16R ATARS modification. The Mission Requirements field of the DRD (Table 6.1.1.1-1) contains the prioritized list of requirements resulting from this examination and was assembled with the support of a word processor. The list was prioritized based on the background of the operational experts with the F-16R mission.

6.1.1.2 System Requirements

The documentation provided by General Dynamics and the specific background of an operational expert helped to determine and prioritize the system requirements for the F-16R in a short period of time. The SORD also described several other system requirements for the F-16R, which were deferred due to funding and technology limitations. The system requirements were documented and prioritized in the System Requirements field of the DRD (Table 6.1.1.2-1).

6.1.1.3 Problem Statement/Cockpit Philosophy

The purpose of this activity was to document a specific set of directives to guide the CDT. For example, this CSDP activity was not developed at the start of the F-16R Project, but a succinct statement was written and documented in the respective fields of the DRD.

a. Problem Statement. The problem statement is as follows and is based on the System Requirements section of the DRD: Improve the PVI design to support ATARS pod functions in the F-16R aircraft design as proposed by General Dynamics.

b. Cockpit Design Philosophy. The cockpit design philosophy is to improve situational awareness and to incorporate automation where possible during imaging, threat avoidance, and/or navigation in order to relieve the high workload environment associated with key parts of the tactical reconnaissance/fighter mission.

Table 6.1.1.1-1. Prioritized Mission Requirements

Rqmt ID	Description
M1	Day-night/under-the-weather imaging
M2	Medium and high altitude stand-off imaging
M3	Threat environments variable (high, medium, or low intensity conflict)
M4	Pilot capable of imaging non-preplanned targets of opportunity
M5	Pilot capable of obtaining imagery while aggressively maneuvering the aircraft
M6	Targets: 4-point targets, 2-point and 1 Line-of-Communication (LOC), or one area target
M7	Pop-up maneuver to medium altitude - initiate collection during climb
M8	Imagery of targets 2 - 5 miles distant (low altitude)
M9	Low altitude daylight scenario: Electro-Optical (EO) sensor primary, dead reckoning techniques and aircraft navigation equipment
M10	Low light scenario: Infrared (IR) sensor primary, Inertial navigation primary
M11	Initial Point (IP) and target imagery (required), waypoints (desired)
M12	Data link of imagery done once in secure area and at safe altitude
M13	Before data link - pilot loads Joint Service Imagery Processing System (JSIPS) coordinates (normally pre flight)
M14	Carriage, launch, and jettison of two Unmanned Aerial Reconnaissance Vehicles (UARVs)

Table 6.1.1.2-1. Prioritized System Requirements

Rqmt ID	Description
S1	Minimize pilot task loading for safe mission accomplishment
S2	Enhance situation awareness and minimize possibility of spatial disorientation
S3	Simplify pilot tasks
S4	Maintain all inherent fighter characteristics
S5	Modified system to be supportable, survivable, and operationally effective
S6	Modify F-16 C/D to a reconnaissance configuration maintaining fighter/reconnaissance dual designed operational capability
S7	Incorporate automated functions where possible
S8	Operational Flight Program (OFP) modifications to host reconnaissance functions on the existing MFD and keyboards/displays and HOTAS to operate ATARS EO sensor suite
S9	Display of sensor video on MFDs
S10	Hands-on control of ATARS sensors and pod functions via HOTAS
S11	Hands-off control of ATARS sensors and pod functions via MFDs
S12	Data entry of JSIPS coordinates
S13	Selection and control of sensor parameters
S14	Display of status in HUD
S15	<u>Deferred</u> System Enhancements: Digital Terrain System
S16	<u>Deferred</u> System Enhancements: Forward Looking infrared (FLIR): (1) off-axis sensor; (2) tank size detect at five nautical miles (NM); (3) head steerable

Table 6.1.1.2-1. Prioritized System Requirements (Continued)

Rqmt ID	Description
S17	Deferred System Enhancements: Helmet Integrated Night Vision System
S18	Deferred System Enhancements: Internal electronic countermeasures
S19	Deferred System Enhancements: ALR-56M Advanced Radar Warning Receiver
S20	Deferred System Enhancements: Missile warning system
S21	Deferred System Enhancements: Automated terrain following
S22	Deferred System Enhancements: Data burst transmission of target data

6.1.1.4 Notional Baseline Cockpit/Cockpit Layout

The purpose of this activity was to functionally and graphically describe the baseline crew system configuration that served as the reference for each iteration of crew system analysis, design, and evaluation. Typically, the procedures require that the CDT first create the configuration from a functional ideas standpoint, then later provide a detailed graphic configuration. However, in this instance, the F-16R Project involved a mature cockpit configuration based on a modified F-16C Block 30.

6.1.1.5 Input to Specifications

The Weapon System Specification (WSS) and the Crew System Specification (CSS) would normally be updated to reflect the impact of recent decisions. These documents were not available and the specific tasks performed did not require access to the documents. However, these tasks will be performed in future demonstrations. The need for specifications becomes paramount in implementing cockpit design for the aircraft system as the design activities of the CSDP are applied. The type of information found in specifications will become the significant driving requirement for a future cockpit product tool to help the CDT contribute to the WSS and CSS.

6.1.1.6 System Drivers

System requirements and several other factors, such as technology attributes, mission tactics, and human performance considerations, must be analyzed to derive the system drivers for the cockpit design. The cockpit system drivers applicable to Field Demonstration No. 1, as listed in Table 6.1.1.2-1, were adapted from the research and development done by General Dynamics. The General Dynamics accomplishments were carefully examined and a number of the drivers were modified slightly based on changes in the program. These drivers were placed in a field of the DRD and will be utilized throughout the remainder of the project activities.

6.1.1.7 Results of Up-Front Analysis

The results of the early efforts were evaluated and project plans were formulated based on the experiences of the CDT and on the subjective review of the preceding activities. Plans were made to perform the implementation aspects for design and simulation activities. Additionally, insight was gained for the planning of analysis activities. The plans are described in Section 6.1.2.

6.1.1.8 Design Traceability Manager

In the above activities, an attempt was made to complete a draft of the results. The product used to house the draft results is the DRD (Reference 22). The DTM is a CDS software tool that, when complete, will allow the results of Up-Front Analysis activities to be directly entered into the DRD. In that way, pertinent requirements are documented for the derived results into a single standalone product. The DRD will be accessible electronically throughout subsequent CSDP activities.

6.1.2 Program Planning

a. Background. Program planning was also accomplished without predefined procedures or tools. It was also accomplished prior to performing the Up-Front Analysis activities. Therefore, the entire planning process under development for the CSDP (Reference 66, Appendix C) was not followed specifically. However, the key elements that could be accomplished without procedures and tools were identified and were applied to the F-16R Project.

b. Progress. Several of the requirements documented in the DRD were taken into consideration and, in lieu of a planning or product tool, an Analysis Study Plan (ASP, Reference 27) was prepared.

The ability to transmit products and data from one CSDP activity to the next, to show traceability, and to verify like data, was of prime importance in planning project activities. A separate Program Planning Tool is required to consolidate the necessary activities, decide on the proper use of the CDT members, and to publish (and maintain) a schedule that reflects up-to-date project activities. The completion of Field Demonstration No. 1 will support the determination of requirements for the Program Planning Tool.

6.1.3 Crew System Analysis

a. Background. At the onset of Field Demonstration No. 1, a determination was made to concentrate on the crew system analysis activities because it was the area where most activities, procedures, and CDS tools were available. The CDT elected to perform only those analyses that had the highest value towards verifying and tracing requirements and baselining the pilot workload. It was also decided to follow the CSDP (Section 4) rather than the activities in the CSDE. This decision was based on the fact that the CSDE did not have detailed procedures to follow. However, an attempt was made to perform activities that are common to both the CSDP and the CSDE and to compare the results. A discussion of the analyses performed is included at the end of this section.

Documenting the results of the analyses has begun. Product Traceability Reports (PTRs) on all activities in the CSDP performed for Field Demonstration No. 1 are currently being documented. Manual recording of the information was necessary because the DTM (which will support the CDT in this area) was not yet fully developed.

b. Progress The DRD was the product of Up-Front Analysis activities, while the Analysis Study Plan was the product of Program Planning. Both documents were required inputs for determining the mission and configuration that set the context for the following Crew System Analysis activities. The context for the analysis activities was the day reconnaissance mission and baseline configuration (denoted as Misn1 and Cfg1, respectively). The overall objective of performing the Crew System Analysis activities was to evaluate the modified F-16C Block 30 for mission effectiveness and pilot performance. Specific objectives are included in Table 6.1.3-1.

Table 6.1.3-1. Crew System Analysis Objectives

Provide focus and scope for design and test and evaluation activities, through recommended critical phases, Action/Information requirements, and preliminary crew system specification.
Develop a model of crew member activities, including functional requirements for identification of baseline pilot performance prediction.
Evaluate the impact of mission functions on pilot capability through task workload analysis.
Develop information requirements dependent on functional requirements for improved designs.
Develop a list of candidate PVI solutions to be rapidly prototyped and evaluated in part-task simulation.
Assess proposed design improvements by comparing the task workload results from the baseline design with those from the proposed design.

Sections 6.1.3.1 through 6.1.3.9 discuss the Crew System Analysis activities performed during this reporting period.

6.1.3.1 Mission Profile Analysis

The purpose of performing Mission Profile Analysis is to generate a graphic depiction of the actual mission events and timing for use in later analysis and evaluation activities. The graphic representation of the mission provides a medium for communication between the Operational and Crew System Analysts, to define a set of critical mission events based on mission requirements. Mission Profile Analysis requires the analysts to define the mission requirements, such as gaming region, and threat/target laydown, so that the flight path can be determined to meet the mission objective, such as collecting pictures of preplanned targets across a road. Once the flight path is determined, the analysts can then identify the timing of critical mission events, such as lethal threat activity or system failures, along with the mission objective event (e.g., target imagery collection) in preparation for such analysis and evaluation activities as functional flow, task workload, action/information, and lo-fidelity simulation.

Input was required from the Mission Requirements and Mission Parameters fields of the DRD and from operational experts to determine how and when each of the critical events developed during Up-Front Analysis should take place in the mission profile. The mission was a tactical reconnaissance profile flown in the Middle East (Persian Gulf). The pilot was tasked to navigate along a preplanned route at a low altitude, and to image reconnaissance targets as the primary task, while reacting to threats. The reconnaissance flight consisted of one F-16R fighter ingressing at a low altitude and high speed in an attempt to locate and record critical information.

The general process employed in performing Mission Profile Analysis consisted of the following:

a. **Identified the mission gaming region**. The gaming region specified for the day reconnaissance mission (Misn1) and baseline configuration (Cfg1) represented the Persian Gulf theater, according to the Design Problem Statement field of the DRD and the operational experts.

b. **Determined threat/target characteristics and location**. Numerous ground threats were to be encountered after crossing the FEBA, especially in and around the target areas. The surface-to-air (SA) threat specifications shown in Table 6.1.3.1-1 were defined by an operational expert according to the Threat Requirements field of the DRD (Reference 22). The threats and targets were defined in MDTOOL and graphically deployed in the gaming region to represent a realistic threat/target laydown.

Table 6.1.3.1-1 SA Threat Specifications

	SA-8	SA-11
Guidance Type	command	semi-active
# Missiles Ready	1	1
Total # Missiles	3	2
Maximum Altitude (feet [ft])	40,000	49,200
Minimum Range (nautical miles [nm])	1	1.6
Maximum Range (nm)	6.5	16.2
Lock to Shoot Delay (seconds [s])	10	10
Reload Time (s)	480	600
Velocity (feet per second)	3346	2900
Sensor Range (nm)	16	20
Antenna Height (ft)	14	10
Antenna Elevation (degrees)	-5 to +85	-5 to +85

c. **Determined aircraft characteristics**. Aircraft parameters were defined for the baseline cockpit, the F-16C Block 30, according to the Baseline Identification and Crew System Design Context fields of the DRD.

d. **Determined aircraft flight path**. The aircraft's route of flight was defined by operational experts through analysis of the scenario, including the mission objectives and the threat/target laydown.

e. **Determined the events that need to take place in the profile**. Using the Need Statement of the SORD, the Mission Requirements, and the Crew System Design Driver fields of the DRD, the day reconnaissance mission was decomposed into fifteen inflight phases (Reference 66, Appendix H). Phase 5 through Phase 10 were identified as those phases critical for meeting mission objectives. The other phases were not chosen for crew system analysis since they represented standard F-16 phases and were unaffected by the support of tactical reconnaissance functions. Several critical events were identified for robustness of the system to meet the objectives of the mission based on the Mission Requirements and System Requirements of the DRD (Reference 66, Appendix H).

f. **Graphically represented the gaming region, threat laydown, target laydown, FEBA, and flight path**. The information defined in Procedures a-e was entered into MDTOOL to build the mission profile file. A graphical representation of the mission profile is presented in Figure 6.1.3.1-1. This figure includes the initial (or start) point, waypoint designations with phase change descriptions noted, timings at selected waypoints, threat locations, and target locations.

g. **Calculated the precise timing, heading, altitude, and airspeed**. The *PG.scn* mission scenario file was executed in MDTOOL to calculate the precise timing, heading, altitude, and airspeed for the route of flight.

h. **Created an ETL to include the precise timing of events**. The ETL was automatically generated by MDTOOL during execution (Procedure g). No new events were added.

The MDTOOL Version 4.05 was used to graphically represent the mission profile. The DMA data of the gaming region was called by MDTOOL. The following options were exercised: creating the FEBA, defining targets, threats and aircraft, and deploying threats/targets and aircraft. MDTOOL provided a sufficient medium for graphically portraying the mission and generating the required product of this activity needed by subsequent activities.

MDTOOL is supported by an Event Definition Data Base. This data base supplies the intervisibility events (e.g., threat # search, lock-on, launch, or crossing waypoints) that occur when the mission profile is executed. The user may also access this data base to customize the ETL resulting from this execution of the mission profile. To date, the user may only select from a small data base of events. The ability to pick points other than waypoints to calculate phases is necessary to accurately layout a mission typical of the aircraft. The user also requires the ability to capture critical events specific to the mission, such as target imagery collection in the case of the F-16R. The Event Definition Data Base is difficult to update and manage to accommodate upgrades. A recommended upgrade to MDTOOL is the ability to directly edit the MDTOOL ETL. This would eliminate the requirement to update the Event Definition Data Base, reducing the configuration management overhead.

The product of this activity was a graphical representation of the mission profile and an ETL (Reference 66, Appendix G). The ETL is required as input for the subsequent Mission Scenario and Functional Flow Analyses. The ETL captures the critical drivers of the mission requirements for determining crew system activities.

6.1.3.2 Mission Scenario Analysis

The objective of Mission Scenario Analysis was to describe, in the form of a written script, the events that would take place in a sequential order during the mission. The mission script

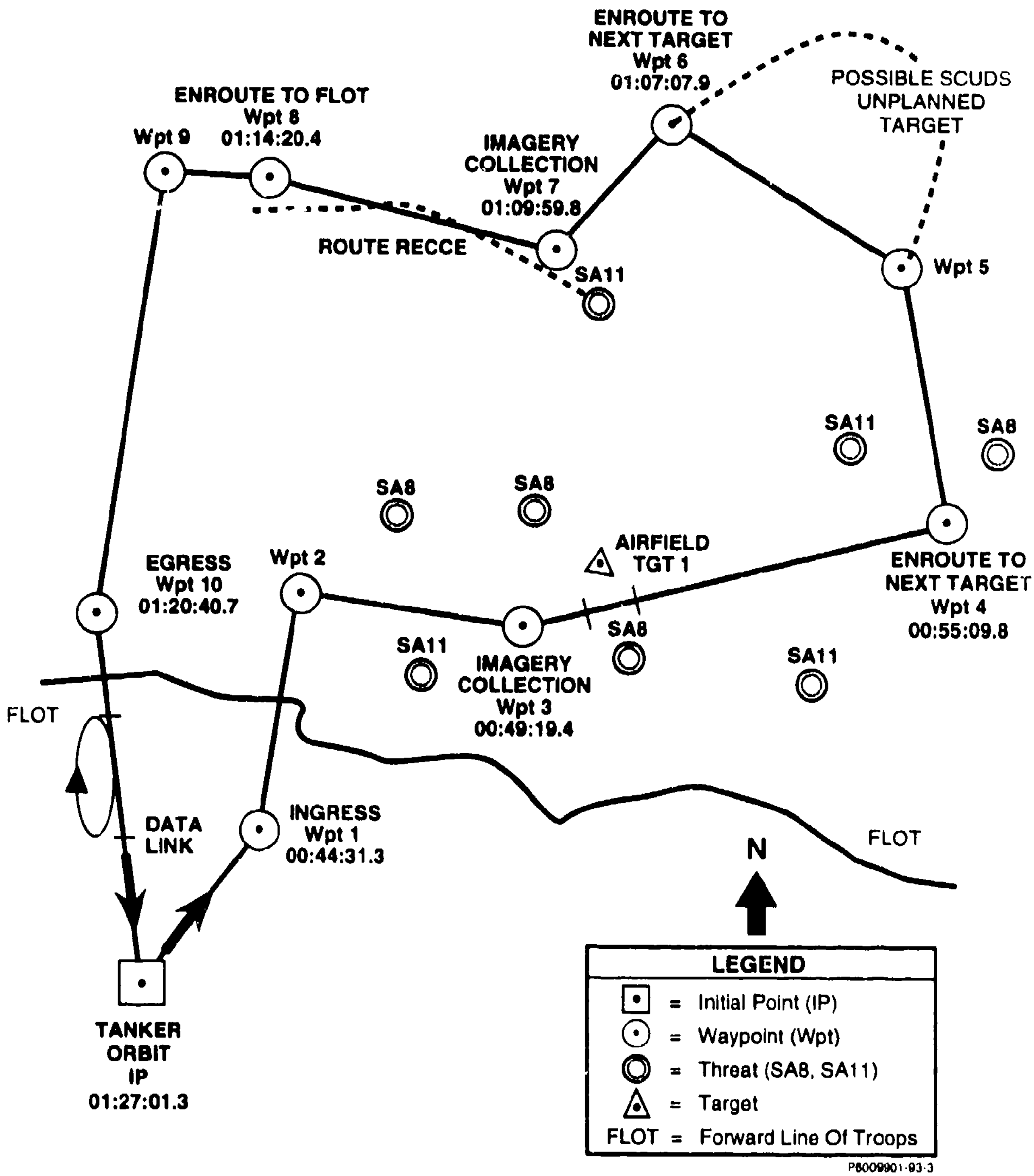


Figure 6.1.3.1-1. Mission Profile

included specific events recorded in detail from the crew member's perspective. The detail includes attributes such as viewing information and actions that took place during each event.

The products of Mission Profile Analysis (graphic profile and ETL) were required as the input for this activity. Further information was obtained from the Mission Requirements field of the DRD. The notional baseline cockpit was also used to derive predictions about crew member's actions with equipment and systems on board.

The general process employed in Mission Scenario Analysis consisted of the following:

a. Wrote script for the mission profile. The mission scenario script information was documented by an operational expert based on the Mission Profile Analysis input and the content of the DRD (Reference 66, Appendix H). Microsoft Word on a Macintosh IIci workstation was used in this activity.

b. Added depth to fully develop or derive lower-level task descriptions and crew interface requirements. Information about each phase was obtained through the use of the concept mapping technique over several sessions. Six sessions were planned; however, numerous follow-up sessions were required for clarification. A white board was used during the mapping sessions and an analyst replicated the white board map on a laptop computer using the TAKE2 software.

c. Ensured that normal, unexpected, and emergency conditions were built into each script. Based on information in the DRD, critical events were defined for the mission profile. The phase-by-phase scripts of activities were reviewed to ensure inclusion of critical events and associated activities, and to ensure that all appropriate normal, unexpected, and emergency conditions and system responses were present. The concept maps from sessions 1 and 2 were used as checklists for normal conditions. Unplanned targets and system failure scripts were also included.

The MDTOOL Version 4.05 was used to view the mission synopsis file that was created with the *vi* editor on the SGI workstation. This same text file was recreated through the use of a Macintosh IIci with Microsoft Word in order to obtain printouts. MDTOOL was also used to view the ETL produced during Mission Profile Analysis.

The product of this analysis was a phase-by-phase written description of events occurring during the mission. This file included normal, emergency, and unexpected conditions that might arise during the flight (Reference 66, Appendix H).

6.1.3.3 Mission Phase/Functional Flow Analysis

Functional Flow Analysis was used to establish the flow of critical mission phases and events and to provide a vehicle for decomposing critical mission events into task-level descriptions of crew member activity. This analysis used critical mission phase Level I, II, and III block diagrams as a means for producing the functional flows. These diagrams are described further in the discussion that follows.

To proceed with Functional Flow Analysis, details about the order of mission events and the type of system responses that needed to take place were required. The output from Mission Profile Analysis (graphic profile and ETL) and Mission Scenario Analysis (mission script) was obtained.

The general process employed in Functional Flow Analysis consisted of the following:

a. Created Level I block diagrams. The Level I block diagram was used to develop the overall flow of the composite mission phases from start to finish. A graphic depiction was created using the TAKE2 software, which is used in conjunction with concept mapping as noted in Section 6.1.1. Figure 6.1.3.3-1 shows an example of a Level I block diagram.

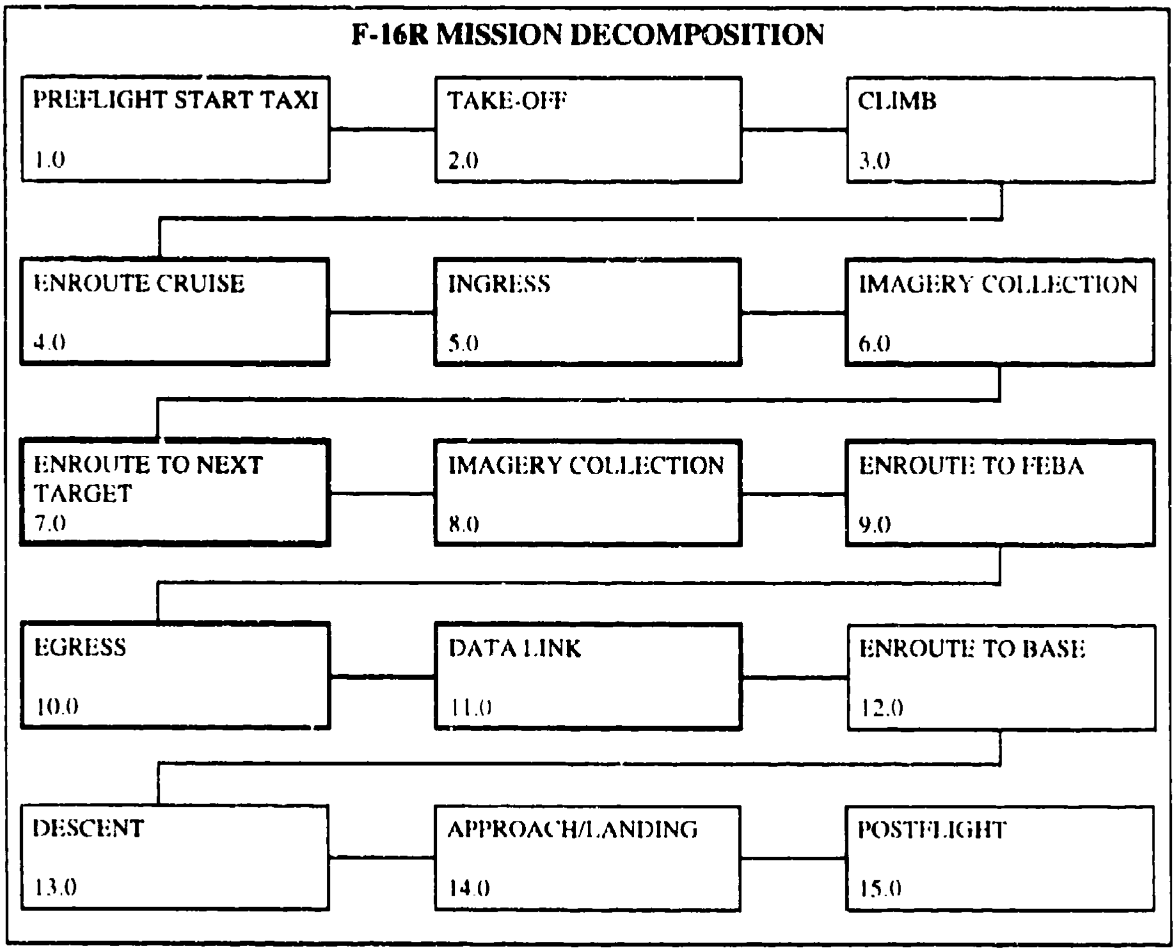


Figure 6.1.3.3-1. F-16R Level I Block Diagram

b. Provided a written description of the events contained in each phase. A Macintosh IIci computer was used to capture the textual descriptions of the mission events (Reference 66, Appendix H).

c. Created Level II block diagrams. The purpose of the Level II block diagrams was to establish the flow of gross task-level (functional) system (aircraft and crew member) activities directly based on the critical mission phases defined in the Level I block diagrams. Figure 6.1.3.3-2 shows an example of a Level II block diagram. In this diagram, blocks are created to capture descriptions of continuous (C) activity in the system during the creation of Level III block diagrams. The format of this diagram implies the sequencing of the functions. A graphic depiction was created using the TAKE2 software, which is used in conjunction with concept mapping as noted in Section 6.1.1.

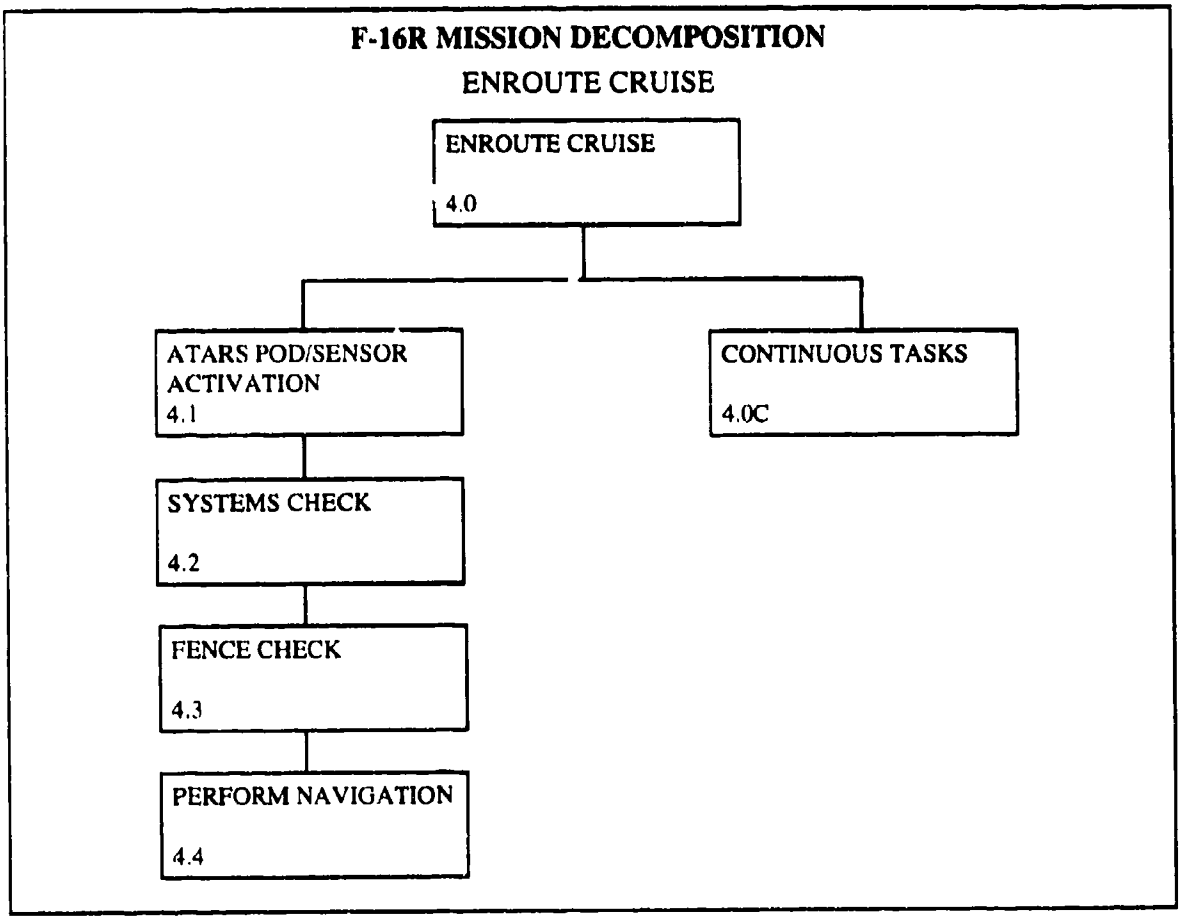


Figure 6.1.3.3-2. F-16R Level II Block Diagram

d. Created Level III block diagrams. The Level III block diagram was developed by further detailing the Level II block diagrams and performing an initial system/pilot task allocation based on a preliminary assessment of the repeatability of the task and the required accuracy of performance. Figure 6.1.3.3-3 shows an example of a Level III block diagram. The format of this diagram denotes the sequence of tasks to be performed. A graphic depiction was created using TAKE2 software, which is used in conjunction with concept mapping as noted in Section 6.1.1.

The Level III block diagrams were the products of Functional Flow Analysis. Originally, approximately 60 concept maps were created using the TAKE2 software after each of the concept mapping sessions.

The TAKE2 software supports the generation of the concept-node-link-concept-node format. After the map is built, the TAKE2 software can generate a data base from the map to help organize and interpret the information contained in the map. A readable form of the data base is

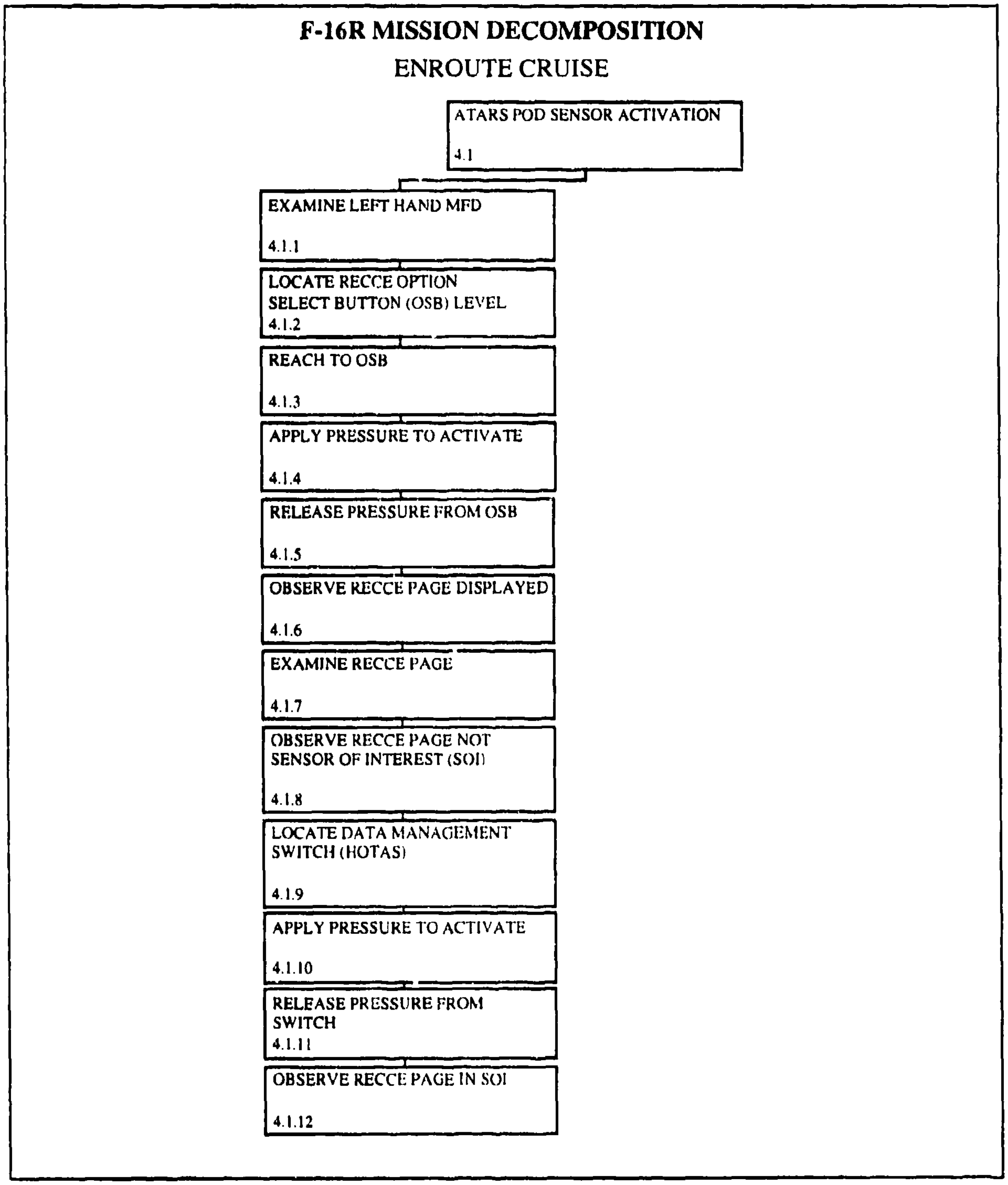


Figure 6.1.3.3-3. F-16R Level III Block Diagram

generated by building an outline of the data base. The TAKE2 software also supports the building of a matrix that allows the user to view concepts in an organized manner across multiple concept maps. The user may define categories and key words for analyzing the informational content of the maps.

The TAKE2 software supported the building of concept maps conveniently during a mapping session, requiring only a portable Macintosh computer. However, the maps required reformatting after each session in order to impose structure that could focus the decomposition of the mission timeline to the task level (Level III block diagrams). The TAKE2 software did not integrate well with documentation (e.g., maps could not be copied and pasted into this document), manageability of data (concept maps are free form and are difficult to get into a structure supported by the graphic capabilities of the TAKE2 software), or with timelines for creating and obtaining printouts. Therefore, these maps were converted into Level I and II diagrams using Powerpoint software.

Level III diagrams were created using Powerpoint software also, but the form was different from the one presented in this section. The Level III block diagrams generated by Powerpoint contained the function block and a listing of all the tasks and associated input parameters for the SWAS Task Workload Analysis software. The reason for this is that Functional Flow and Task Workload Analyses were occurring almost in parallel due to time limitations. After reviewing the Level III maps created that included all of the task workload input data, it was determined that readability was greatly inhibited and that the maps should be reformatted for this document. The results assisted in clarifying the format of the product from Functional Flow Analysis.

The Level I, II, and III maps created in Powerpoint can be found in Appendix I (Reference 66). The original maps are located in the F-16R Crew System Analysis Notebook (Reference 28). The TAKE2 outliner was attempted unsuccessfully. This was attributed to the fact that a Beta version of the software was being used. The TAKE2 matrix capability was not exercised for supporting Functional Flow Analysis.

6.1.3.4 Action/Information Requirements Generation

According to the CSDP, in order to develop the controls and displays for a cockpit, an Action/Information Requirements Analysis should be accomplished. This activity will generate specific displayed information or control requirements that must be implemented into the cockpit design. Each task must be broken out into the finite elements of the information that the crew member needs prior to making a decision or taking an action, followed by the necessary interface characteristics of how to control the associated action. The CDT should perform this analysis prior to completing the re-design of selected controls and displays for the enhanced cockpit of the F-16R. This analysis will be completed using a data base and will be implemented for the re-design effort.

6.1.3.5 Task Descriptions

The purpose of this activity was to translate the output of Functional Flow Analysis (Level III block diagrams) into input data for the TTL analysis and workload analysis. The output Task Data Base (generated by TMT when fully developed) contains the Level III block diagram task title, a description of crew member activities within the context of the mission and events, and a list of associated discrete tasks necessary to complete the task description in action-verb/object format.

The general process employed in developing task descriptions consisted of the following:

a. Decomposed lowest level of functional flow into task descriptions. A task description was created for each of the Level III block diagram lowest-level nodes. The task descriptions contained information on how an individual crew member would perform the task at that point in the mission.

b. Laid out what each task entailed for completion. The list of tasks were reviewed to ensure that all actions to be performed by the crew member were included, taking into account each possible channel involved (i.e., visual, auditory, psychomotor, and mental).

The product of this activity currently is a SWAS file containing the information specified by the functions and sub-functions. The task description file provides the necessary input for performing TTL Analysis. This activity was performed in the SWAS data base file, since TMT was not implemented in time to perform this activity. TMT will provide a much smoother transfer of data and will allow manipulation of the data base for further analysis. The full set of data from this analysis can be found in Appendix J (Reference 66).

6.1.3.6 Functional Allocation Trade-off

In a typical crew system analysis set of activities, functional allocation tradeoff analysis should be performed to determine what tasking should be accomplished by whom or by what subsystem. With the F-16R Project, this analysis activity was approached differently. Given that the baseline was a nearly complete cockpit with a single crew member and fully defined interfaces, the CDT was not required to perform tradeoff analysis. However, when the enhanced version of the F-16R cockpit is examined, functional allocation trade-off analysis will be performed using the QFD methodology. The QFD Designer will be evaluated during this analysis.

6.1.3.7 Task Timeline Analysis Generation

The purpose of performing the Task Timeline (TTL) Analysis is to ensure that all of the crew member task requirements are addressed for the baseline cockpit design. Detailed information about the channels required to perform the task (that is, visual, auditory, psychomotor, and mental), physical aspects of each task such as reach distances, types of reach, visual attributes, forces, releases, rotation angles, and accurate and appropriate time values for each task, were assessed and compiled. The output data was used as the basis for Task/Workload Analysis.

The task description file created during the development of the task descriptions was required as input to TTL Analysis. The general process employed in performing TTL Analysis consisted of the following:

a. Assessed baseline cockpit design attributes. The mechanization of the F-16R Block 30C was reviewed using the mechanization document created by General Dynamics (Reference 29). Controls and displays were identified for the performance of each task description.

b. Defined requirements for each task. The sequence of discrete tasks required to complete the task description were reviewed (from the Task Description File) and verification was made of the individual levels of channel activity.

c. Defined physical aspects of the tasks. The reach distances, categories of reach used, visual scanning angles and distances, speech requirements, positioning, displacements, forces, releases, rotation angles, etc., were defined for each task in order to correctly assess time requirements associated with task execution. Accurate geometry data is essential in defining the physical aspects of the tasks. Given an electronic format containing the geometry data, an approach can be taken using a tool such as COMBIMAN to derive reach distances. A more labor-intensive approach is to derive the data directly from engineering drawings. The approach taken by the CDT was to take measurements from the F-16 cockpit simulator in the USAF Aeronautical Systems Center CSEF. This simulator was an actual F-16 cab so the measurements should be correct. These measurements (manually documented), coupled with the expertise of operational analysts and the CDT personnel familiar with F-16 operations, enabled the CDT to correctly define all aspects of both the normal F-16 tasks and the added reconnaissance tasking.

d. Determined accurate and appropriate time values for each task. The data from the F-16 measurement activity and principles of Methods-Time Measurement (MTM) were used to predict pilot movement distance times. The SWAS (the commercial software used to support workload analysis) includes an MTM module that allows an operator to derive MTM time data for each task. While this feature is beneficial for tasks with unique physical aspects, it is cumbersome to apply to a large list of tasks with similar physical aspects. The CDT developed a listing of basic motions required and applied the principles of MTM, to develop a look-up table of information that provided the CDT with a more efficient means to enter time values into the SWAS data base (Reference 66, Appendix K).

e. Developed and completed data base of task timing. The output of this activity was an update to the TTL data base (currently residing in SWAS) and an input for the Task/Workload Analysis activity. Since SWAS does not have the capability to import data files, the task data were entered directly into SWAS. Channel activity, physical aspects, task precedence, and timing requirements were entered. The SWAS internally calculated appropriate timing factors to assist in the workload calculations.

Normally, the CDT would execute these activities separately to ensure that the proper attention was given to each task description and associated parameters when determining task times prior to estimating workload; however, the tools required (TMT and a workload tool with the ability to import the TTL data base) were not available for use.

6.1.3.8 Task/Workload Analysis

The objective of performing Task/Workload Analysis was to assess the capability of the pilot to complete the intended mission of the F-16R as defined in the day reconnaissance mission scenario script. The F-16C Block 30 controls and displays and the additional controls and displays required to support operational use of the ATARS pod were used as the baseline configuration. The following discussion describes the methodology employed to conduct the analysis, the results of the analysis, the conclusions drawn from the results, and the tool used for analysis. A full report is located in Appendix J (Reference 66).

The focus of the Task/Workload Analysis was on the impact of the discrete tasks associated with the use of the ATARS pod. Continuous task time and processing allotments were inserted into each mission phase to account for non-essential inter-aircraft coordination and routine aircraft flight control and system management.

The Task/Workload Analysis was based on the Mission Profile, Mission Scenario, and Functional Flow Analysis results. The Mission Scenario script described the activity conducted during the execution of the day reconnaissance mission, while the Level III Functional Flow block diagrams provided the necessary sub-function information for deriving task descriptions.

The Task/Workload activities and their related procedures were under development during the time that the scheduled Task/Workload Analysis was to be performed. The Functional Flow Analysis was complete but the Task Description and Task Timeline Analysis activities were incomplete. Therefore, the Task Description and Task Timeline Analysis activities were performed during Task/Workload Analysis. In future field demonstrations, the analysis activities will be accomplished in the proper CSDP sequence.

The Task Description and Timeline data were compiled for assessing operator workload. The SWAS was used in support of this effort to generate blocks of common operator tasks, to timeline those tasks, to merge task blocks into full mission segments, and to analyze mission segments using Monte Carlo simulation techniques. The SWAS analysis results are reported in the form of a summary of descriptive statistics based on the discrete and continuous task time requirements, and the time available for the completion of the segment tasks.

The contents of a typical portion of a mission segment summary are described in Table 6.1.3.8-1. It includes the segment number and section title, the workload data (95% confidence interval [CI] and mean workload factor), the time requirements (95% confidence interval and mean time requirements), the time available and the probability for successfully completing the mission segment in the allotted time. The fields contained in the summary are further described below.

Table 6.1.3.8-1. Mission Segment Summary

Segment Number: Four
Section Title: Waypoint Six to Overfly Update

<u>Summary of Results</u>:

<u>Workload</u>:	
95% CI-Lower Bound:	0.55
95% CI-Upper Bound:	0.57
Mean Workload:	0.56
<u>Time Requirement</u>:	
95% CI-Lower Bound:	232.51
95% CI-Upper Bound:	255.42
Mean Time Required:	246.51
<u>Time Available</u>:	289.00
<u>Probability of Success</u>:	100%

Workload Estimate. This is an estimate of the pilot's workload during the completion of the mission segment. The workload estimate is based on the ratio of time available to time required, and thus represents the percentage of time when the pilot is occupied. The workload estimate considers the time requirements associated with the completion of the discrete tasks and the continuous tasks. In the above example, Mean Workload of .56 means that the crew member was tasked for 56% of the time (over the trials performed in the model), and that the time required to perform the tasking is less than the time available. Therefore, this section of the mission should be accomplished with relative ease, as compared with a higher workload percentage segment.

Time Requirements. This is a break down of the time required to accomplish the discrete tasks assigned to the pilot.

Time Available. The time available reported here is extracted from the mission analysis and the timeline associated with flight from one location to another.

Probability of Success. This provides an estimate of the potential for completing the required tasks in the allotted time available.

The results for each of the individual analyses conducted in support of this assessment are reported in the F-16R Initial Mission Task and Workload Analysis report (Reference 66, Appendix J). All results were determined using the data generated from this analysis. Assessments were made based on the data, as well as expert interpretations of the data, and only summary report data is presented in this report. Table 6.1.3.8-2 summarizes the workload estimates for each critical phase of the Baseline F-16R Mission.

Table 6.1.3.8-2. Summary Critical Phase Workload Estimates: Baseline F-16R Task/Workload Analysis

Phase Number	Phase Title	Section Number	Mean Workload
Four	Enroute Cruise	One	0.80
Five	Ingress	One Two	0.92 **1.00**
Six	Imagery Collection	One Two	**1.00** **1.12**
Seven	Enroute to Next Target	One Two Three	0.97 0.69 0.87
Eight	Imagery Collection	One	0.63
Nine	Enroute to FEBA	One	0.67
Ten	Egress	One	0.72
Eleven	Datalink	One	0.75

Bold = Workload Exceeding Time Available

Those analysis results suggest that the F-16R pilot may experience periods of excessive workload with low probability of successful completion at critical times. This is based on the mission requirements, mission equipment package, projected tasking requirements, and potential threat/flight environment.

Review of the tasks associated with the baseline mission indicated that, in many instances, the contributing cause for excessive workload was the cumulative effect of the basic F-16C Block 30 control and display interface. When applied to the execution of this mission, the recommended future automation features incorporated to the F-16C Block 30 design would reduce pilot workload and contribute to a successful mission.

Individual sections of the Workload Analysis Report (Reference 66, Appendix J) address some of the potential automation crew coordination and control/display integration candidates. These candidates included automation of waypoint selection (as next steerpoint), incorporation of a

Global Positioning System (GPS) receiver as a means of updating the inertials, integrated controls and displays, a Tactical Situation Display (TSD), and a Helmet-Mounted Display (HMD), and consideration of a two-place cockpit to offload pilot tasking to a weapon systems officer or similarly trained personnel.

From the analysis, consideration of a two-place aircraft could be supported in that, during low-level ingress, the pilot was fully occupied with head-up tasking (given there was no Terrain Following (TF) or Terrain Avoidance (TA) coupled autopilot mode), and that the ATARS interface was strictly head-down. Another option would be to consider making the ATARS interface primarily head-up (e.g., use HOTAS control inputs and provide head-up imagery by the incorporation of a helmet-mounted display). Another option might be to use an HMD with flight and threat symbology to complement the ATARS head-down activity.

One topic that was explored during this initial assessment was the impact of threat activity in the vicinity of the objective (target imagery). In the phases with the highest levels of workload, the pilot was largely occupied with monitoring and responding to the presence of ground-based threats.

Given the distraction due to the presence of threats, it is likely that the pilot would reduce the time delegated to tasks associated with the preparation of the ATARS equipment and use the time instead to attend to the threats. This would logically seem to have a negative impact on successful mission completion. Again, delegation of the ATARS-specific tasks to a second crew member would free the pilot to respond to the existing threats. Another alternative is to integrate information onto a single TSD so that much less visual scanning time would be needed. Another benefit of the TSD is that it would provide a better planning capability if threats and mission profiles were integrated with other tactical information on a single cockpit display.

Based on the results of this initial assessment, several issues should be addressed. A one-versus two-man crew comparison, an HMD-based ATARS interface versus the proposed head-down interface and/or an integrated TSD, are all candidates for an F-16R cockpit redesign. Additional trade-off studies should be accomplished to assess the potential for incorporating automation as a means of relieving the pilot of some of the routine tasking in the event that the single-seat aircraft is retained.

6.1.3.9 Alternative Analysis Activities

Although the CSDP was followed during Functional Flow Analysis, a decision was made to compare the CSDE Crew System Analysis activities and software tool support to the CSDP and tool support. The objective was to start with identical top-level input and follow both methodologies (i.e., the CSDP versus the CSDE) for translating mission requirements into a model of crew member activities (decomposition of an ETL into a TTL). Then, a check was made to ensure that the same tasks were being modeled according to the proper dwell time and channel activity. Finally, the two different task/workload models were invoked and the results compared.

The following activities during the F-16R Phase Six (Imagery Collection) were performed. The initial ETL was imported from MDTOOL. This timeline contained phase transitions and threat/target information. Using the ETL and concept mapping to decompose the mission, timelines were devised to accommodate the tasks populating the CWAS data base. The forced four-level decomposition (i.e., Event, Function, Procedure, and Task) structure demanded the use of the Event-Function and Function-Procedure Relationship Data files.

The general process employed during this alternative analysis consisted of the following:

a. Performed Function Analysis. The ETL (generated previously) and the existing EFR data file were reviewed. Using the descriptions in the concept maps, functions from the EFR were mapped to concept map descriptions. These functions were then attached to each event using NMT, resulting in a Function Timeline (FTL).

b. Selected/Defined Comparability Baseline Crew Station. The F-16C Block 30 was specified by the Crew System Mechanization Document (Reference 30) and the DRD as the modified F-16R aircraft. The F-16 cockpit geometry data in the CAD format, required for the development of a TTL and input for the original analysis software, was unavailable to meet CCCD schedule requirements. The closest cockpit configuration available through the Cockpit Geometry Data Base was the CAT Design aircraft that was modeled after a number of fighter type aircraft (F-4, F-15, F-16). This file is called the F-16R .CIPLP.CCCIN file.

Zones were identified for the purposes of housing panels and providing a 2D plane in space. Panels were identified for the purpose of grouping or housing the controls and displays. Extraneous panels were removed. The F-16R_CIPLP.CCCIN file was reviewed and updated to remove inapplicable controls and displays, and F-16R specific controls and displays were assigned to the respective panels. The impact of this work-around will not be realized until the results of the workload analysis are assessed and compared to both the SWAS and the evaluation (EDSIM) results.

c. Generated Control/Display Catalog. Dwell time data that was specific to the various controls and indicators used in the F-16R during Phase Six were modified. For this comparative analysis, the dwell times were entered to directly relate to the dwell times developed for SWAS using the MTM techniques.

d. Performed Procedure Analysis. The FTL (generated previously) and the existing FPR data file were reviewed. Using the descriptions in the concept maps, procedures from the FPR were mapped to concept map descriptions. These procedures were then attached to each function using NMT, resulting in a Procedure Timeline (PTL).

e. Defined Tasks for each Procedure. Referencing the concept maps (Level II and III block diagrams), the TTL was constructed using the modified F16R_CIPLP.CCCIN file. While viewing a PTL and the F16R_CIPLP.CCCIN file in NMT, tasks were built for each procedure, using the proper controls and indicators from the F16R_CIPLP.CCCIN file. These tasks corresponded to the discrete tasks from the SWAS data base.

All functions and procedures were assigned start times corresponding to the parent event start time. For this analysis, these times were shifted within the event duration. An input file for the Procedure-Level Timeline Analysis Software was written by NMT (*.MSAIN file) after the TTL was completely developed. The file included channel of activity, control/display or pseudo-device to perform task, and appropriate start and stop times.

f. Performed Workload Analysis. The Procedure-Level Timeline Analysis Software was executed on the TTL input file and output plots were generated. The interpretation of the results and evaluation of the methodology according to the criteria is currently in progress.

The TTL was the product of the original CDS tools. The TTL was a four-level treenet structure with a standardized taxonomy for text descriptions of events, functions, procedures, and tasks. The procedures and tasks were the basis for the Procedure- and Task-Level Timeline Analysis set of software. A printout of the complete TTL can found in Appendix L (Reference 66).

6.1.4 Crew System Design

a. Background. The activities normally conducted for the Crew System Design section of the CSDP are based on the design - evaluate - redesign premise in which all results are carefully examined prior to deciding on the next design implementation. Because this is a demonstration of the still maturing CSDP, only a few design and evaluation iterations will be performed.

b. Progress. The Action/Information Requirements Analysis will be performed to ensure that the enhanced control/display design will trace each requirement to the pilot's informational or control need. Although that analysis is not complete, re-designing has begun with some of the baseline implementation of the F-16R through rapid prototyping with the GMS software tool. This expedites the design and evaluation of cockpit displays and controls. This type of activity acts as a coarse filter to identify the most promising control and display concepts for further development and evaluation.

6.1.5 Crew System Evaluation

a. Background. One of the advantages of following a process, in which activities are dependent on earlier tasks being accomplished, is that it gives team members time to understand and plan more effectively how best to evaluate requirements and design. In the case of the F-16R, the simulation test plan was prepared when few results were available. This was done to assist in the development of the EDSIM, which is also in the early development stage.

Normally on a project of this type, the analyses would have been partially completed prior to creating the first simulator evaluation. This program performed analysis and simulation in parallel to accommodate the CDT personnel and schedule requirements. An attempt was made to feed much of the analysis results to the test planning process.

b. Progress. This section presents the result of the optimization between what would normally be evaluated on a program tempered against the timetable and what was possible to put together as a demonstration. The test plan was developed to structure a study that is typical of what can be reasonably accomplished in the cockpit design environment. The test will be implemented with only a few subject pilots since, as is generally the case in cockpit design, schedules, resources, and pilot availability limit the number of pilots in any given study except where the criticality of the PVI design decision warrants otherwise. The entire system will be put through the normal checkout and evaluation phases to see what can be accomplished with this type of part-task simulation activity.

The actual evaluation of all demonstration events is underway with the testing of several analysis and design activities that were discussed in earlier sections. The results of these and the upcoming EDSIM evaluation will bring valuable information to the area of process activities, activity procedures, and tool requirements. The data predicted in the analysis phase will be verified to learn if it is comparable to that collected with actual pilots in the evaluation.

On 11 May 1993, a draft version of the test plan was submitted for review. Comments were received and updates are being implemented. However, the updated version (Reference 31) of the plan was completed in this reporting period. The following is a summary of the main contents of the test plan. In keeping with the general philosophy of test plans, emphasis is given to the detailed plans for conducting the tests rather than to the rationale behind those details.

The purpose of the test plan is to provide the documentation necessary to specify the nature of the test, test objectives, methodology, and related information, with enough detail so that (1) all approval authorities will be able to make timely decisions and (2) implementation of the test plan can commence. This test plan was developed to serve as a guidance document for conducting the F-16R study and to identify the strong and weak points of the CDS as related to pilot-in-the-loop evaluation studies.

As part of the test plan development process, inputs were provided by six SMEs with extensive operational experience in air-to-air combat, air-to-ground attack, tactical reconnaissance, and the ATARS system. Human factors inputs were provided by personnel having extensive experience with pilot-in-the-loop simulations and field test environments.

The primary objectives of the test are to (1) compare a defined baseline configuration F-16R, including an integrated ATARS pod, with a configuration comprised of an improved PVI; (2) obtain results for comparison with those that were analytically derived; (3) obtain pilot performance and workload data for populating analytical data bases; and (4) provide insights into the effectiveness of CSDP and tools for conducting cockpit evaluations.

The EDSIM is the simulator that will be used in this study. Support personnel for conducting this study, and the subject pilots, are provided for under the contract. Descriptions of the baseline and enhanced configurations to be compared are described in the test plan. The primary differences center on the enhanced configuration containing a horizontal situation display (HSD), an automatic target hand-off system (ATHS), and global positioning system (GPS) data in the cockpit.

The performance of this evaluation is controlled by the schedule of activities and the state of development of the EDSIM. Therefore, it is being treated as a demonstration rather than a formal experiment. As such, the number of factors being tested versus the data collected and analyzed will be judged and reported accordingly. In the performance of this activity, procedures will be outlined for all areas of the evaluation demonstration. These procedures will become the first draft for the CSDP implementation. This activity is anticipated to begin during June and end in July, 1993.

6.1.6 Conversion of Engineering Design Simulator

a. Background. The Breadboard Cockpit Simulator (BSIM) was built during the CAT Program to support anthropomorphic and human factors testing on various cockpit configurations. The BSIM has a slightly different focus than many simulators in its class. It is intended to be used during the evolution of a cockpit design to reflect changes in both the physical and operational characteristics of an evolving cockpit. Such changes include the orientation of the seat, the consoles, the stick and throttle, and the displays and controls to reflect an evolutionary cockpit design. The name of the BSIM was changed to the EDSIM to emphasis its engineering development role. However, its purpose remains essentially the same, which is to support pilot-in-the-loop simulation in a rapidly reconfigurable cockpit.

For Field Demonstration No. 1, the CDT is configuring the EDSIM to represent an F-16C/D Block 30 cockpit (Figure 6.1.6-1). The objective for the layout was not to provide the full capabilities of a complex domed simulator, but rather to incorporate a limited number of master modes, cursor controls, and related features in order to integrate controls and displays, and to simplify display sensor management. The cockpit controls and displays required for the F-16R Project are described in the following paragraphs. A full description of cockpit implementation requirements will be available when the Crew Station Mechanization Document (Reference 30) is finalized.

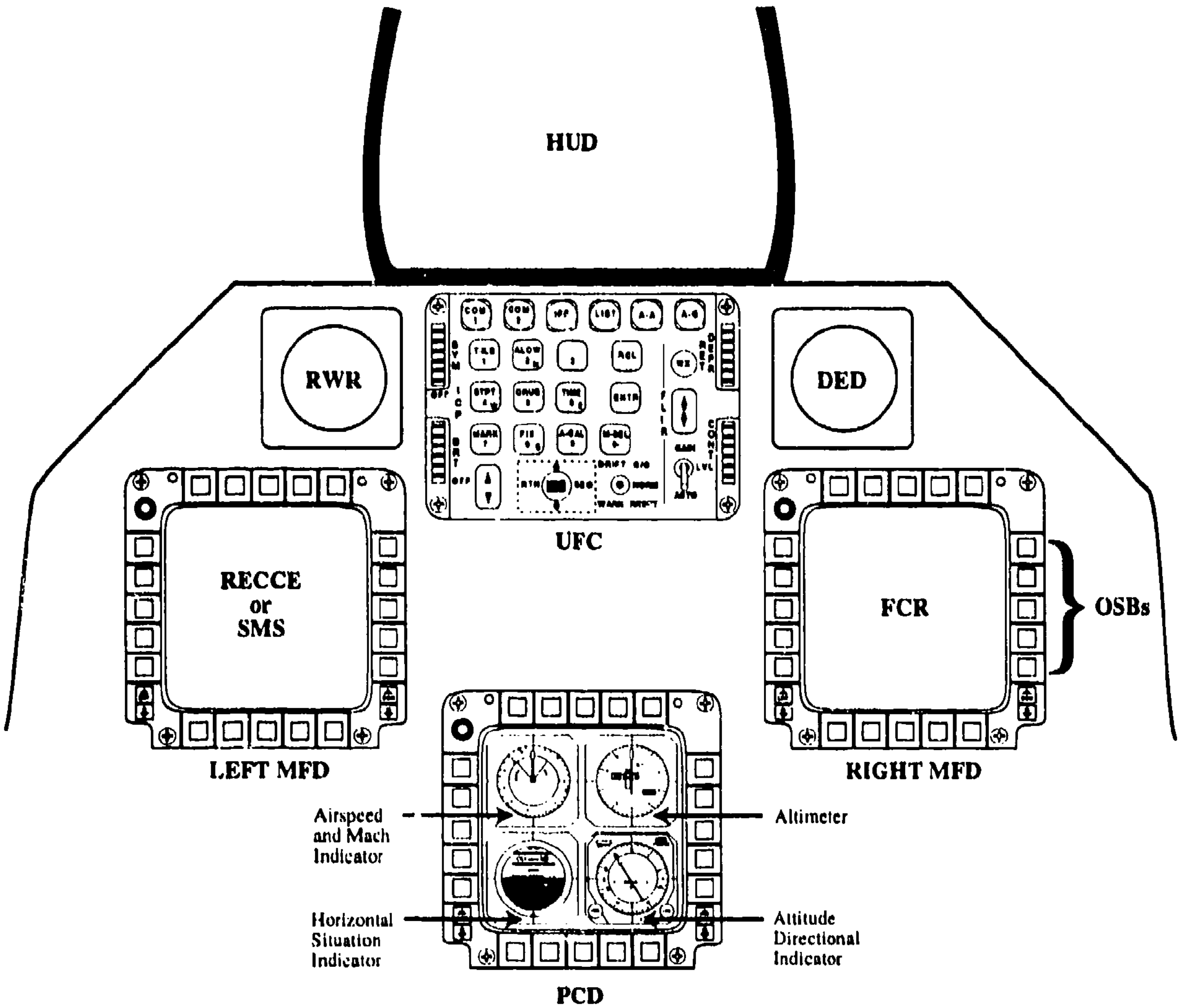


Figure 6.1.6-1. EDSIM Cockpit Layout for the F-16R Project

The HUD is generic in design and is presented on a 19-inch color monitor mounted above the instrument panel. This same color monitor provides the pilot with a sixty-four-degree, low-fidelity out-the-window view.

The Radar Warning Receiver (RWR) and the Data Entry Display (DED) are replicated on 4x4-inch MFDs. The RWR display is generic in nature and provides the pilot with ground threat location, radar tracking, and missile launch indications. The UFC and the DED are not mechanized in the EDSIM because the simulator response times for this hardware exceed several seconds and adversely affect simulation conditions.

The Left and Right MFDs measure 6x6 inches versus the 4x4-inch displays found in the actual aircraft. The Left MFD is programmed to display the SMS or Recce Format Pages. The Right MFD is programmed to display the Fire Control Radar (FCR).

The Airspeed and Mach Indicator, Altimeter, HSI, and the ADI, have been replicated electronically on a 6x6-inch display. This composite EDSIM display is called the Performance/Control Display (PCD), and it is presented on the MFD that is located below and between the Left and Right MFDs.

The console panels are positioned in approximately the same location as in an F-16 C/D. Console switches do not represent the actual switches located in the aircraft and are not functional. Only those Option Select Buttons (OSBs) that are required for minimum tasking/objectives are functional, i.e., Auto/Manual/Override (OSB#2) or Sensor Select (OSB#18).

b. Progress. Veda personnel visited the ASC CSEF to examine its F-16 simulator as a basis for defining the necessary changes to the EDSIM. The decision was made to mechanize the F-16R Left and Right MFDs by using two of the three existing MFDs in the EDSIM. SGI workstations v1sg11 and v1sg13 will be used as the respective MFD display processors running GMS programs. The v1sg15 workstation will be used to drive the F-16 HUD, and the v1sg17 to drive the out-the-window external scene. The v1sg14 workstation will drive two Special-Purpose Displays (SPDs) for the Threat Display and the Up Front Controller output. The Performance/ Control Display (PCD) will be driven by the v1sg9 workstation. The v1sg16 workstation will be used to display the ATARS sensor output and would be overlaid on the Left MFD using v1sg11.

The panels on the console were reconfigured into position based on the F-16 drawings and using the panels and monitors that will provide the needed realism during Field Demonstration No. 1. The Molex connectors and associated wires to connect the F-16 stick and throttle hardware were purchased.

The Airspeed, MACH, ALT, AFI, and HSI gauges of the PCD are operating correctly. The MFD pages for the first field demonstration were developed based on the needs for the part-task, part-mission test being demonstrated. Seven reconnaissance pages, three SMS pages, and one Master format page, were developed for the MFDs. The ATARS view will be implemented using Merit's MAGIK SCENE and will be overlaid on the MFD pages using the GMS tool. The MFD development is underway. The right MFD will be developed by modifying the existing, hard-coded MFDs. The GMS design of the PCD took approximately six-and-a-half days to complete; the display driver and the data generator programs took two days to develop.

6.1.6.1 Stick and Throttle

a. Background. The EDSIM at the outset of the contract did not include F-16 hand controls. Emphasis was placed on obtaining an F-16 C/D Block 40 stick and throttle in order to accurately capture the essential elements of the HOTAS mechanization required for the F-16R mission.

b. Progress. The F-16 Block 40 side stick controller and throttle grip (Figure 6.1.6.1-1) were acquired from Technology Products Incorporated after comparison with devices from other suppliers. The HOTAS mechanization includes three active switches on the throttle grip: the Dogfight Switch, the Speed Brake Switch, and the Cursor/Enable Switch. The HOTAS mechanization includes two active switches on the stick: the CMS Switch and the Trim Button.

After delivery, the F-16 throttle was deemed to be inaccurate, and the vendor was contacted to determine if a more geometrically correct device could be fabricated. The vendor committed to a 3-4 week turnaround, once adequate data was received. Arrangements were made for the vendor to take photographs of the CSEF throttle to assist in the modeling. In addition, a source for borrowing an actual F-16 throttle was located in the 4950TW Fabrication Modification Division. Arrangements were made to borrow the throttle as a master form for molds.

6.1.6.2 Aero Model

a. Background. CATBATS provides the choice of operating any aircraft as either a six-degree-of-freedom (6-DOF) flight model, or a 3-DOF flight model. The 6-DOF model uses twelve state variables to describe the aircraft motion. Three rotational angles and three angular rates describe the aircraft's attitude, while three position and three velocity variables describe the aircraft's translational motion. The 6-DOF model uses variable flaps, slats, landing gear, and atmospheric effects (e.g., layered winds), and offers the capability to represent a full flight from takeoff to landing. The 6-DOF model also uses a dual engine model for thrust. Each engine can be individually controlled, but both engines will lie on the centerline of the aircraft. The 6DOF aero and engine models are table-driven and can be reconfigured by modifying the tables.

The 3-DOF model uses only the translational motion to describe the aircraft. This model does not use any of the other features previously listed for the 6-DOF model and is usually used with the threat aircraft.

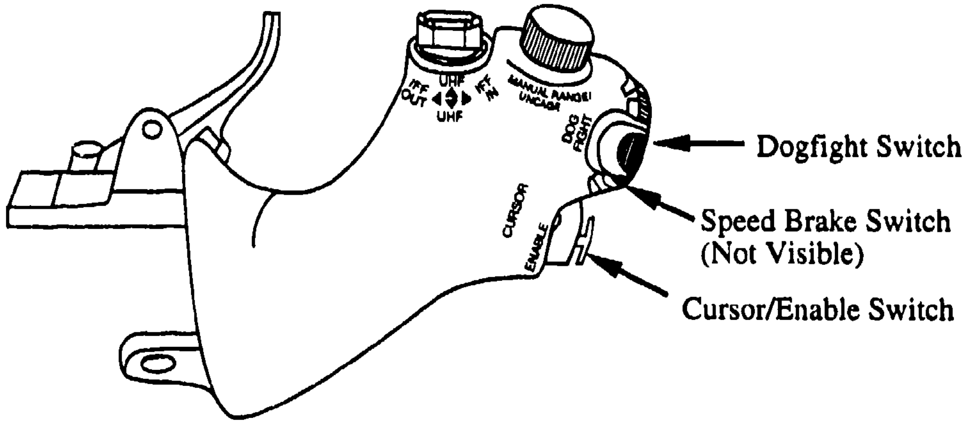
Merit Technology studied the possibility of integrating an F-16 aerodynamic model acquired from Williams AFB into BATS. To integrate this model, an Interface Control Document would be needed and a common block to hold the data would have to be generated. This required more time than was available before the first field demonstration. If F-16 data (coefficients for the airframe, engine and weapon flyouts) can be generated, it will be relatively straightforward to modify these data files to run BATBIRD with F-16 performance.

b. Progress. An aeronautical engineer from Veda was consulted to research and validate the flight coefficients that are used in the BATBIRD model. It was discovered that there were a number of missing coefficients and that these could be supplied, along with the required coefficients, for the F-16. The evaluation revealed that a related and significant problem exists with transport delays and/or latency in the system. This problem is being investigated further.

A method to validate the F-16 aerodynamic coefficients was established by arranging to have the existing coefficients extracted for comparison with F-16 truth data. The EDSIM dynamics were found to be detuned significantly. Tests run with actual F-16 data showed that detuning was necessary for stable flight in the EDSIM.

The Merit Technology representative assisted in modifying a copy of the out-the-window code to adapt it to the ATARS sensor. There was a problem with the out-the-window imagery pausing when data was being scrolled. This problem was investigated and an FDR file was sent to

Throttle Grip



Side Stick Controller

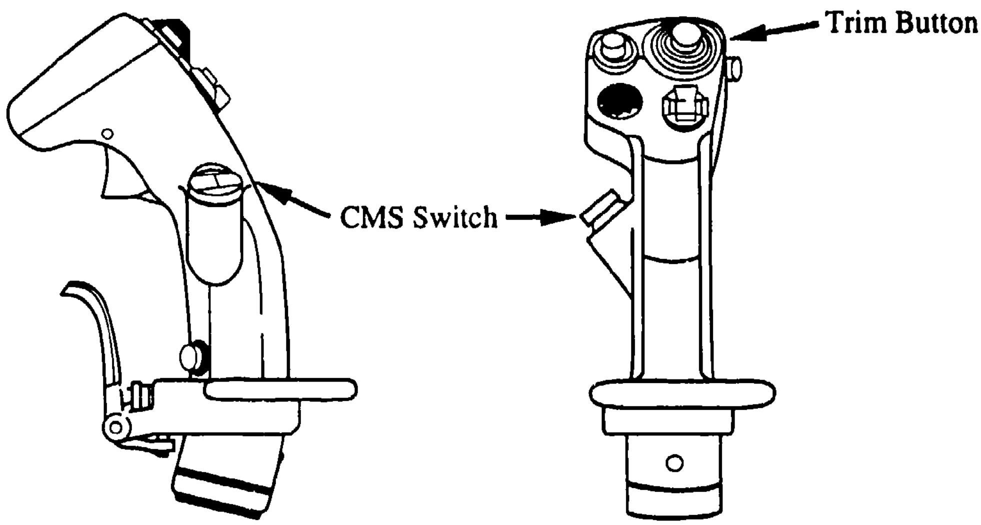


Figure 6.1.6.1-1. EDSIM Hands-on Throttle and Stick

Merit for further study. Merit modified the shell scripts and the environment variables to allow MDTOOL and CATBATS to view the same directories.

6.1.6.3 Workload Assessment Monitor

a. Background. The WAM (Workload Assessment Monitor, Reference 22) is a standalone system developed by the Armstrong Laboratory. WAM measures physiological data (for example, heart rate, eye blink, and respiration) which can be used to help understand the causes of pilot workload. The data gathered by WAM will be correlated to the EDSIM output by clock timing using the WAM host computer, an IBM PC-compatible processor.

b. Progress. The data gathered by the WAM is presently collected and processed internally. To integrate this data into the EDSIM system, wall-clock time must be correlated to simulator time. The following method was used to verify the usefulness of the WAM.

A single pulse was sent to the WAM to begin recording and correlating simulator time to wall-clock time in the EDSIM. This was accomplished by recording the wall-clock time in the FDR file. To assess and analyze the workload measured by WAM, the analyst must know what events are taking place. In addition, a specific mission scenario (either existing or to be created) will need to be used to test the WAM.

6.1.6.4 Tones and Intelligent Input/Output Control Boards

a. Background. The Intelligent Input/Output Controller (IIOC) includes a number of analog and digital input/output interface boards, a controller board, and a Direct Memory Access (DMA) interface to the BSIM Interface Controller (BIC). The IIOC provides the interface from the EDSIM-mounted interface consoles (i.e., switches, dials, lights, stick, and throttle) to the simulation software. The Tones Board provides alerting or status tones to the pilot during pilot-in-the-loop simulations.

b. Progress. Veda recommended that the Tones and IIOC boards be removed from the EDSIM configuration to eliminate the BIC, an underutilized piece of hardware. When the boards are removed from the BIC, the BIC will no longer be needed for the EDSIM that is installed in the laboratory, nor for any units that may be installed in the field. Elimination of the requirement for the BIC will save on the cost of each initial installation if this system is fielded.

The Tones and IIOC boards were relocated to the vlsg16 and vlsg17 workstations, respectively. Both boards were integrated, tested, and successfully verified.

6.2 Human Subjects Research

a. Background. Research in the C-CADS Laboratory will expose personnel to known, controlled risks associated with the use of the following items: (a) CDS computers and peripherals; (b) general equipment and supplies; (c) the EDSIM; (d) the WAM; and (e) the building enclosure. The potential risks imposed by each of these items are discussed below.

(1) CDS Computers and Peripherals. The CDS workstations consist of unmodified commercial off-the-shelf computer systems that are connected via commercial networks to printers, plotters, data storage devices, and other workstations. Connections to the EDSIM are made through commercial networks also. These items comply with the best commercial practice in terms of electrical shock protection, electromagnetic radiation shielding, physical injury protection, and

ergonomic design. Cabling is adequately protected and secured to keep walkways clear. Workstations and work surfaces are solid, stable, and ergonomically sound.

(2) General Equipment and Supplies. The equipment and supplies consist of expendable items (disks, tapes, printer cartridges, paper, etc.) that are commercially obtained and commonplace in the office environment. The risk of exposure to hazardous materials is minimal.

(3) EDSIM. A reconfigurable part-task simulator, the EDSIM consists of an adjustable metal framework that supports commercial and custom electronic components used to emulate cockpit devices. A formal analysis of the hazards imposed by the EDSIM is warranted but has not been done yet. A preliminary assessment indicates the following potential sources of risk:

(a) Physical injury during the subject's normal ingress or egress from the EDSIM due to possible contact with unprotected metal edges and corners, the lack of designated handholds, and/or tripping over cables that are not fully secured or properly positioned. These risks could be magnified by the variable-intensity room lighting. To minimize these risks, the room lighting will be returned to its full intensity to ensure adequate lighting during subject ingress and egress from the EDSIM.

(b) Emergency ingress and egress from the EDSIM imposes the same risks identified above, but potentially exacerbated by haste and the potential inadequacy or obscuration of emergency lighting. CCCD Program Office personnel evaluated the emergency lighting previously and installed additional lighting; thus these risks appear to be managed.

(c) The McFadden hydraulic system will not impose a risk because it will not be used for Field Demonstration No. 1. The EDSIM cockpit display devices will impose a controlled risk because they consist of commercial off-the-shelf display devices, connected to the CDS processors through commercial networks. These items comply with accepted commercial practice in terms of electrical shock protection, electromagnetic radiation shielding, and physical injury protection, although a more thorough assessment is warranted.

(4) WAM. Veda has been informed that an earlier configuration of the WAM passed a formal risk analysis, but that the WAM configuration in the C-CADS Laboratory contains different amplifiers that have not yet been formally assessed. Thus, there is a possibility that the WAM could expose the subject to electrical shock if it is improperly used or if a catastrophic failure occurred. These risks are minimized by the fact that the WAM was designed by AL/CFH personnel to meet established safety criteria, and installed under their supervision. The Veda personnel who will be operating the WAM have been trained in its proper use.

(5) Building. Room 109 of Building 248 offers ample egress and fire equipment. Air supply, air exchange, and heating/cooling systems are fully satisfactory. Facility construction, wiring, and plumbing comply with accepted commercial practice. Lighting intensity is variable through wall-mounted rheostats. Lighting will be returned to full intensity during EDSIM ingress and egress. Subjects will not be confined to the facility, but will be free to leave at any time.

User activity is consistent with job responsibilities as members of the CDS support contract. The ability to gainfully apply the CDS tools and process will not be a factor in maintaining present employment.

b. Progress. A formal safety inspection is required to address the above mentioned factors, and to correct deficiencies. Subject to these conditions, the conclusion of the above risk assessment is that the risks incurred by the CDS users will be adequately managed during the validation testing process, and there are no reasonable expectations of serious physiological or psychological injury from the testing.

7. REFERENCE LIST

1. Department of the Air Force, Air Force Materiel Command. *Statement of Work: Crew-Centered Cockpit Design (CCCD), Field Demonstration Program.* Contract No. F33615-92-C-5936. Wright-Patterson Air Force Base, OH: AFMC: 28 Sep 1992 .

2. Chartier, Allen and others. *Quality Function Deployment Designer User's Reference.* Dearborn, MI and West Bloomfield, MI: American Supplier Institutes, Incorporated and Qualisoft Corporation; 1991

3. Mitta, Deborah A. *An Application of the Analytical Hierarchy Process: A Ranking-Ordering of Computer Interface.* Santa Monica CA: Human Factors and Ergonomic Society; Mar 1993.

4. Zaff, Brian S. and others. *Capturing Multiple Perspectives: A User-Centered Approach to Knowledge and Design Acquisition.* Dayton, OH: Logicon Technical Services, Incorporated; 1990.

5. Sequitur Systems, Incorporated. *Sequitur's Workload Analysis System User's Manual, Version 4.0.* Forth Worth, TX: Sequitur Systems, Incorporated; 15 Mar 1991.

6. Lehman, Edward and others. *Configuration Management Plan for the Cockpit Design System.* Contract No. F33615-88-D-0533, Document No. 63209-92U/P6760. Dayton, OH: Veda Incorporated; 11 Mar 1992.

7. Air Force Software Technology Support Center. *Conference Tutorials.* Hill Air Force Base, UT: Air Force Software Technology Support Center, HQ USAF/SC; 15 Apr 1991.

8. Air Force Software Technology Support Center. *Requirements Analysis and Design Tool Reports.* Hill Air Force Base, UT: Air Force Software Technology Support Center, HQ USAF/SC; 15 Apr 1991.

9. Snyder, D. and others. *Draft User's Guide for the Concept Analysis (Beta Version 1.0).* Dayton, OH: Research Design Incorporated; Aug 1992.

10. Adkisson, S.M. and others. *Display Development Study for the Cockpit Automation Design Support System (Sherrill-Lubinski's GMS and VPI's VAP Evaluation).* Report No. D56-100301-1. Seattle, WA: Boeing Advanced Systems; 1989

11. Veda Incorporated. *Cockpit Design System (CDS) Support Procedures, Final Report for Delivery Order 9.* Contract No. F33615-88-D-0533, Document No. 63659-92U/P6760. Dayton, OH: Veda Incorporated; 31 Dec 1992.

12. Visual Edge Software Limited. *UIM/X 2.0 Developer's Manual, Document Version 4.92.* Los Altos, CA: Visual Edge Software Limited; 1991.

13. Visual Edge Software Limited. *UIM/X 2.0 Developer's Manual, Document Version 8.92.* Los Altos, CA: Visual Edge Software Limited; 1991.

14. Veda Incorporated. *Final Report: Analyze and Assess the CAT Cockpits.* Contract No. F33615-88-D-0533, Delivery Order 10, Document No. 63665-92U/P6762. Dayton, OH: Veda Incorporated; Dec 1992.

15. Veda Incorporated. *Upgrade Plan for the Cockpit Design System.* Contract No. F33615-92-C-5936, Document No. 63060-93U/P60099. Dayton, OH: Veda Incorporated; 15 Jan 1993.

16. Anderson, A.F. and White, R.W. *System/Segment Specification for the Cockpit Automation Design Support System.* Contract No. F33615-86-C-0537. Seattle, WA: Boeing Military Airplanes; 10 Jun 1991.

17. Garcia, Lucy. *Design Traceability Manager (DTM) Design Document.* Contract No. F33615-92-C-5936, Document No. 63112-93U/P60099. Dayton, OH: Veda Incorporated; 30 Jul 1993.

18. Martin, Cynthia D. *Trip Report - McDonnell-Douglas - Tactical Aircraft Mission Planning Systems (TAMPS).* Contract No. F33615-92-C-5936, Document No. 63823-93U/P60099. Dayton, OH: Veda Incorporated; 30 Oct 1992.

19. Lehman, Edward. *Validation Test Plan, Field Demonstration No. 1, F-16R Manned Reconnaissance Cockpit.* Contract No. F33615-92-C-5936, Document No. 63771-93U/P60099. Dayton, OH: Veda Incorporated; 16 Jul 1993.

20. Veda Incorporated. *Assessment Report.* Contract No. F33615-88-D-0533, Document No. 63715-92U/P6760 . Dayton, OH: Veda Incorporated; 30 Oct 1992.

21. Allen, S. R. and others. *DEN Specification.* Contract No. F33615-86-C-0537, Document No. D561-10056-1. Seattle, WA: Boeing Military Planes; 1991.

22. Martin, Cynthia. *Draft Design Requirements Document for the F-16 Manned Reconnaissance System.* Contract No. F33615-92-C-5936, Document No. 63198-93U/P60099. Dayton, OH: Veda Incorporated; 4 Feb 1993.

23. Martin, Cynthia. *Final Timeline Management Tool (TMT) Design Document.* Contract No. F33615-92-C-5936, Document No. 63110-93U/P60099. Dayton, OH: Veda Incorporated; 30 Jul 1993.

24. Martin, Cynthia D. *Final Geometry Interface Tool (GIT) Design Document.* Contract No. F33615-92-C-5936, Document No. 63111-93U/P60099. Dayton, OH: Veda, Incorporated; 30 Jul 1993.

25. Storey, Brett. *Final Report, Analyze and Assess the CAT Cockpit (Appendix G, Part II: Workload Tools Trade Study).* Contract No. F33615-88-D-0533, Delivery Order 10, Document No. 63665-92U/P6762. Dayton, OH: Veda Incorporated; 10 Dec 1992.

26. United States Air Force, Headquarters Tactical Air Command. *System Operational Requirements Document for a Follow-on Manned Reconnaissance System.* TAF 302-87-I/II-A, Rev 1. Sanitized version available, originally classified as Secret. Langley Air Force Base, VA: United States Air Force, Headquarters Tactical Air Command; 30 May 1991.

27. Martin, Cynthia D. *Draft Analysis Study Plan for the F-16 Manned Reconnaissance System.* Document No. 63199-93U/P0099. Dayton, OH: Veda Incorporated; 18 Feb 1993.

28. Martin, Cynthia D. *Working Draft Crew System Analysis Notebook.* Contract No. F33615-92-C-5936, Document No. 63824-93U/60099. Dayton, OH: Veda Incorporated; Jan 1993.

29. General Dynamics, Fort Worth Division. *Avionics System Mechanization of the F-16R with the Advanced Tactical Air Reconnaissance System (ATARS).* Document No. 16PR9149. Fort Worth, TX: General Dynamics; Aug 1992.

30. Baltzer, Robert. *Crew System Mechanization Document.* Contract No. F33615-92-C-0533, Document No. 63759-93U/P60099. Dayton, OH: Veda Incorporated; 1993.

31. Roleck, Evan and Boone, Andrew. *Draft F-16 Reconnaissance Test Plan, CCCD Field Demonstration No. 1.* Contract No. F33615-92-C-5936, Document No. 63728-93U/P60099. Dayton, OH: Veda Incorporated; 14 Jul 1993.

32. Wilson, Glenn F. and others. *Workload Assessment Monitor User Manual.* Wright-Patterson, AFB, OH: Armstrong Laboratories; 1993.

33. Amell, J. R. and Stollings, M. N. *Users Manual for the Cockpit Automation Design Support System, Volume 3: Mission Decomposition Tool.* Contract F33615-86-C-0537, Document No. D561-10044-3. Seattle, WA: Boeing Military Planes; 10 Sep 1991.

34. Dempster, D. and others. *Program Maintenance Manual for the Cockpit Automation Design Support System, Volume 3: Mission Decomposition Tool.* Contract F33615-86-C-0537, Document No. D561-10045-3. Seattle, WA: Boeing Military Planes; 10 Sep 1991.

35. Givens, Bret R. and Hoering, Timothy R. *A Scalable Architecture for the Rapid Prototyping of Aircraft Cockpits.* Proceedings of the National Aerospace Electronics Conference; Dayton, OH: Institute of Electrical and Electronic Engineers; Catalog No. 93CH3306-8.

36. Harding, D. P. *The Human Factor for Computer-Aided Design, Volume 3, Document No. 5.* Wright-Patterson Air Force Base, OH: Crew System Ergonomics Information Analysis Center (CSERIAC); 1992.

37. Lehman, Edward and others. *Configuration Management Plan for the Cockpit Design System.* Contract No. F33615-92-C-5936, Document No. 63717-93U/P60099. Dayton, OH: Veda Incorporated; 18 Jun 1993.

38. Lehman, Edward. *Maintenance and Licensing Plan.* Contract No. F33615-92-C-5936, Document No. 63095-93U/P60099. Dayton, OH: Veda Incorporated; Sep 1993.

39. Allen, S. R. and others. *CAT Program Maintenance Manual for the Cockpit Automation Design Support System, Volume 6: Mission Simulation Tools.* Books 6A-6J, Contract No. F33615-86-C-0537, Report No. D561-10045-6. Seattle, WA: Boeing Military Planes; 1991.

40. Allen, S. R. and others. *CAT Users Manual for the Cockpit Automation Design Support System, Volume 6: Mission Simulation Tools.* Contract No. F33615-86-C-0537, Report No. D561-10045-6. Seattle, WA: Boeing Military Planes; 1991.

41. Allen, S.R. and others. *CAT Final Report, CAT Design Process, Volume IIA: CAT Methodology, Production and Development.* Contract No. F33615-86-C-0537, Report No. D561-10059-2. Seattle, WA: Boeing Military Planes; 1991.

42. Allen, S. R. and others. *CAT Final Report, CAT Design Process, Volume IIB: CAT Methodology, Concept Definition.* Contract No. F33615-86-C-0537, Report No. D561-10059-2. Seattle, WA: Boeing Military Planes; 1991.

43. Allen, S. R. and others. *CAT Final Report, CAT Design Process, Volume IIC: CAT Methodology, Demonstration and Validation.* Contract No. F33615-86-C-0537, Report No. D561-10059-2. Seattle, WA: Boeing Military Planes; 1991.

44. Allen, S. R. and others. *CAT Final Report, CAT Design Process, Volume IID: CAT Methodology, Full-Scale Development.* Contract No. F33615-86-C-0537, Report No. D561-10059-2. Seattle, WA: Boeing Military Planes; 1991.

45. Allen S. R. and others. *CAT Final Report, CAT Design Process, Volume IIE: CAT Methodology, Production and Deployment.* Contract No. F33615-86-C-0537, Report No. D561-10059-2. Seattle, WA: Boeing Military Planes; 1991.

46. Asente, Paul J. and Swick, Ralph R. *X-Window System Toolkit, The Complete Programmer's Guide and Specification X Version 11, Release 4.* Bedford, MA: Digital Equipment Corporation; 1990.

47. General Dynamics, Fort Worth Division. *F-16 C/D Avionics Systems Manual, Tape 3.* CDRL Item 28043, Contract No. F33657-84-C-0247. Fort Worth, TX: General Dynamics; Apr 1991.

48. Informix Software, Incorporated. *INFORMIX ONLINE Administrator's Guide Database Server, Version 5.0.* Menlo Park, CA: Informix Software, Incorporated; Sep 1992.

49. Informix Software, Incorporated. *INFORMIX 4GL by Example Data Base Tool, Version 4.10.* Menlo Park, CA: Informix Software, Incorporated; Jul 1991.

50. Informix Software, Incorporated. *INFORMIX Guide to SQL Reference.* Menlo Park, CA: Informix Software, Incorporated; Dec 1991.

51. Informix Software, Incorporated. *INFORMIX 4GL Database Tool, Volume 1, Version 4.00.* Menlo Park, CA: Informix Software, Incorporated; Mar 1990.

52. Informix Software, Incorporated. *INFORMIX 4GL Database Tool, Volume 2, Version 4.00.* Menlo Park, CA: Informix Software, Incorporated; Mar 1990.

53. Informix Software, Incorporated. *INFORMIX Error Messages.* Menlo Park, CA: Informix Software, Incorporated; Dec 1991.

54. Informix Software, Incorporated. *INFORMIX ESQL/C Programmer's Manual, Database Tool, Version 5.0.* Menlo Park, CA: Informix Software, Incorporated; Aug 1992.

55. Karger, Delmar W. and Bayha, Franklin H. *Engineered Work Measurement: The Principles, Techniques, and Data of Methods-Time and Motion Study, and Related Applications Engineering Data, 3rd Edition.* New York, NY: Industrial Press, Incorporated; 1977.

56. Myers, Debbie. *IRIS Showcase User's Guide.* Mountain View, CA: Silicon Graphics, Incorporated; 1991.

57. Open Software Foundation. *OSF/Motif Programmer's Guide, Revision 1.1.* Englewood Cliffs, NJ: PTR Prentice Hall, Incorporated; 1991.

58. Open Software Foundation. *OSF/Motif Programmer's Reference, Revision 1.1.* Englewood Cliffs, NJ: PTR Prentice Hall, Incorporated; 1991.

59. Open Software Foundation. *OSF/Motif Styles Guide, Revision 1.1.* Englewood Cliffs, NJ: PTR Prentice Hall, Incorporated; 1991.

60. Scheifler, Robert W. and others. *X-Windows System, The Complete Reference X Version 11, Release 4, Second Edition.* Bedford, MA: Digital Equipment Corporation; 1990.

61. Structural Dynamics Research Corporation. *Introduction to-I-DEAS.* Milford, OH: Structural Dynamics Research Corporation; 1991.

62. Structural Dynamics Research Corporation. *I-DEAS Core Utilities User's Guide Supplement.* Milford, OH: Structural Dynamics Research Corporation; 1991.

63. Structural Dynamics Research Corporation. *I-DEAS Pearl Menu Guide.* Milford, OH: Structural Dynamics Research Corporation; 1991.

64. Structural Dynamics Research Corporation. *I-DEAS Pearl User's Guide Supplement.* Milford, OH: Structural Dynamics Research Corporation; 1991.

65. Structural Dynamics Research Corporation. *I-DEAS Pearl User's Guide.* Milford, OH: Structural Dynamics Research Corporation; 1991.

66. Veda Incorporated. *Volume II, Interim Technical Report No. 1 (ITR1), Appendices A through L.* Contract No. F33615-92-C-5936, Document No. 63819-93U/P60099. Dayton OH: Veda Incorporated; Dec 1993.